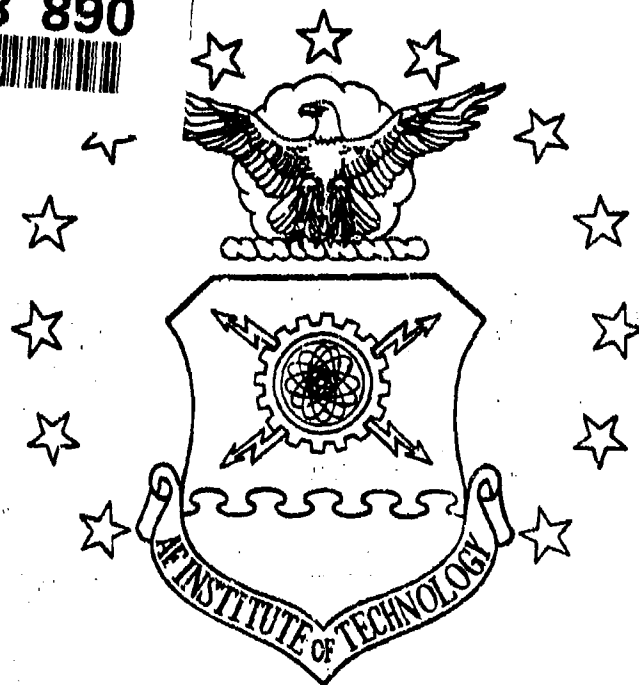
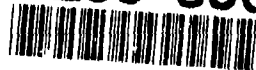


AD-A238 890



DTIC
FI ECTE
JUL 22 1991
S D

A NAVIGATION REFERENCE SYSTEM (NRS) USING
GLOBAL POSITIONING SYSTEM (GPS)
AND TRANSPONDER AIDING

THESIS

Richard D. Stacey
Captain, USAF

AFIT/GE/ENG/91M-04

DISTRIBUTION STATEMENT A

Approved for public release
Distribution Unlimited

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY (ATC)
AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT/GE/ENG/91M-04

DTIC
ELECTE
JUL 22 1991
S D D

A NAVIGATION REFERENCE SYSTEM (NRS) USING
GLOBAL POSITIONING SYSTEM (GPS)
AND TRANSPONDER AIDING

THESIS

Richard D. Stacey
Captain, USAF

AFIT/GE/ENG/91M-04

Approved for public release; distribution unlimited

91 7 19 165

91-05738



REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE March 1991		3. REPORT TYPE AND DATES COVERED Master's Thesis	
4. TITLE AND SUBTITLE A PROPOSED NAVIGATION REFERENCE SYSTEM (NRS) USING GLOBAL POSITIONING SYSTEM (GPS) AIDING				5. FUNDING NUMBERS	
6. AUTHOR(S) Richard D. Stacey, Captain, USAF					
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Air Force Institute of Technology WPAFB OH 45433-8583				8. PERFORMING ORGANIZATION REPORT NUMBER AFIT/GE/ENG/91M-04	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) 6585th Test Group / GDAN Holloman AFB, NM 88330-5000				10. SPONSORING / MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES					
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution unlimited				12b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words) To quantify the performance abilities of existing or proposed navigation systems, the U.S. Air Force has for the last several years compared the performance of the system under test to the performance of a baseline navigation system known as the Completely Integrated Reference Instrumentation System (CIRIS). CIRIS obtains a highly accurate navigation solution by combining the output from three major subsystems: inertial navigation system (INS) information, barometric altitude information, and range and range-rate data from ground transponders which have been precisely surveyed. Although the navigation solution produced by CIRIS is highly accurate, it will soon be inadequate as the standard against which future navigation systems can be tested. This research proposes an alternative to CIRIS - a hybrid Navigation Reference System (NRS) which is designed to take advantage of a newer INS (the LN-93), certain features of the current CIRIS, and certain features of the Global Positioning System (GPS). Analysis is conducted using a Kalman filter development package known as the Multimodel Simulation for Optimal Filter Evaluation (MSOFE). Both a large order truth model for the NRS (in which a full 24 satellite constellation is modeled) and a full-order Kalman filter are developed. Results suggest that the proposed NRS (with GPS aiding) provides a significantly improved navigation solution as compared to CIRIS.					
14. SUBJECT TERMS Navigation Reference System (NRS), Global Positioning System (GPS), Inertial Navigation System (INS), Completely Integrated Reference Instrumentation System (CIRIS), Kalman Filtering				15. NUMBER OF PAGES 259	
				16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL		

GENERAL INSTRUCTIONS FOR COMPLETING SF 298

The Report Documentation Page (RDP) is used in announcing and cataloging reports. It is important that this information be consistent with the rest of the report, particularly the cover and title page. Instructions for filling in each block of the form follow. It is important to *stay within the lines* to meet optical scanning requirements.

Block 1. Agency Use Only (Leave blank).

Block 2. Report Date. Full publication date including day, month, and year, if available (e.g. 1 Jan 88). Must cite at least the year.

Block 3. Type of Report and Dates Covered. State whether report is interim, final, etc. If applicable, enter inclusive report dates (e.g. 10 Jun 87 - 30 Jun 88).

Block 4. Title and Subtitle. A title is taken from the part of the report that provides the most meaningful and complete information. When a report is prepared in more than one volume, repeat the primary title, add volume number, and include subtitle for the specific volume. On classified documents enter the title classification in parentheses.

Block 5. Funding Numbers. To include contract and grant numbers; may include program element number(s), project number(s), task number(s), and work unit number(s). Use the following labels:

C - Contract	PR - Project
G - Grant	TA - Task
PE - Program Element	WU - Work Unit Accession No.

Block 6. Author(s). Name(s) of person(s) responsible for writing the report, performing the research, or credited with the content of the report. If editor or compiler, this should follow the name(s).

Block 7. Performing Organization Name(s) and Address(es). Self-explanatory.

Block 8. Performing Organization Report Number. Enter the unique alphanumeric report number(s) assigned by the organization performing the report.

Block 9. Sponsoring/Monitoring Agency Name(s) and Address(es). Self-explanatory.

Block 10. Sponsoring/Monitoring Agency Report Number. (If known)

Block 11. Supplementary Notes. Enter information not included elsewhere such as: Prepared in cooperation with...; Trans. of...; To be published in.... When a report is revised, include a statement whether the new report supersedes or supplements the older report.

Block 12a. Distribution/Availability Statement. Denotes public availability or limitations. Cite any availability to the public. Enter additional limitations or special markings in all capitals (e.g. NOFORN, REL, ITAR).

DOD - See DoDD 5230.24, "Distribution Statements on Technical Documents."

DOE - See authorities.

NASA - See Handbook NHB 2200.2.

NTIS - Leave blank.

Block 12b. Distribution Code.

DOD - Leave blank.

DOE - Enter DOE distribution categories from the Standard Distribution for Unclassified Scientific and Technical Reports.

NASA - Leave blank.

NTIS - Leave blank.

Block 13. Abstract. Include a brief (*Maximum 200 words*) factual summary of the most significant information contained in the report.

Block 14. Subject Terms. Keywords or phrases identifying major subjects in the report.

Block 15. Number of Pages. Enter the total number of pages.

Block 16. Price Code. Enter appropriate price code (*NTIS only*).

Blocks 17. - 19. Security Classifications. Self-explanatory. Enter U.S. Security Classification in accordance with U.S. Security Regulations (i.e., UNCLASSIFIED). If form contains classified information, stamp classification on the top and bottom of the page.

Block 20. Limitation of Abstract. This block must be completed to assign a limitation to the abstract. Enter either UL (unlimited) or SAR (same as report). An entry in this block is necessary if the abstract is to be limited. If blank, the abstract is assumed to be unlimited.

AFIT/GE/ENG/91M-04

A NAVIGATION REFERENCE SYSTEM (NRS) USING
GLOBAL POSITIONING SYSTEM (GPS)
AND TRANSPONDER AIDING

THESIS

Presented to the Faculty of the School of Engineering
of the Air Force Institute of Technology
Air University

In Partial Fulfillment of the
Requirements for the Degree of
Master of Science in Electrical Engineering

Richard D. Stacey, B.S.E.E.
Captain, USAF

6 March 1991

Accession For	
NHS	CRAGI
DEB	1983
U.S. OCEAN	LI
J. A. DeHon	
By	
Distribution	
Avail. Policy Codes	
Dist	Avail. and/or Special
A-1	

Approved for public release; distribution unlimited



Preface

This document represents a great deal. To some extent, it represents the personal accomplishment of individual research. However, a more fundamental fact is apparent. This thesis represents the educational opportunities and superb research facilities which have been available to me at AFIT. Closer scrutiny reveals a still more basic fact. This thesis represents the outgrowth of my affiliation with the highly professional AFIT faculty and with my classmates here. Ultimately, however, this document represents the *years* of dedicated research which preceded it.

In reviewing the draft of this document, I noted that the references (which have proven to be vitally important to me) collectively span *twenty years*. This is a particularly important fact in my estimation. First it demonstrates that some "timeless" works have been produced which underpin all research of this type. Additionally, it points out how strongly current developments can affect the direction and emphasis of those engaged in independent research such as this.

I do not claim (to borrow the immortal words of Newton) that I have "seen farther than other men", but in my attempt to see beyond my *own* horizons, I have certainly "stood on the shoulders of giants." Those whose influence has contributed to the work represented here may never know it, but I wish to thank them profoundly for their effort. My respect for their vision, my appreciation of their dedication, and my outright amazement at their accomplishments have all grown without bound during my tour at AFIT.

I wish to extend *special* thanks to my thesis committee members, Dr Peter Maybeck and Col Stan Lewantowicz, and to my thesis advisor, Capt Randy Paschall, for their contributions to this research. Their collective experience and advice proved to be an extraordinarily rich resource which never diminished in quality or quantity.

I also want to thank Captains Britt Snodgrass and Jack Taylor. Their selfless assistance throughout this effort proved invaluable in helping me to "find the handles."

In a somewhat sentimental tradition of my own, I want to include the names of my classmates here. Their contributions have come in many forms - not always academic. Let friends be friends forever.

Tom Cox, Ken Crosby, Roger Evans, Tom Flynn, Maurice Martin, Russ Miller, Rob Moyle, Barbara Niblett, Paul Rohs, Dan Sims, Jack Taylor, Mark VonBokern, Dave Wheaton

I will remember all of you, always, and I wish you the *best*.

Though the contributions of those mentioned above were many, and the magnitude of my appreciation is great, there is one who has contributed far more than they, and to whom I offer my deepest gratitude. I will not compare her gentle, understanding nature to others, for none can compare. She will always be the light and happiness in my life. I dedicate this work to my devoted wife, Valerie, and to our unborn child she bears so proudly. I love them both beyond measure.

Richard D. Stacey

Table of Contents

	Page
Proface	ii
Table of Contents	iv
List of Figures	ix
List of Tables	xvi
Abstract	xvii
 I. Introduction	 1-1
1.1 Background	1-1
1.2 Problem Statement	1-2
1.3 Summary of Applicable Previous Research	1-2
1.3.1 GPS-INS Integration	1-2
1.3.2 GPS-CIRIS Integration	1-3
1.3.3 CIRIS Kalman Filter Improvements	1-4
1.4 Research Objectives	1-4
1.5 Assumptions	1-5
1.6 Research Approach	1-6
1.7 Computer and Software Usage	1-7
1.8 Summary of Planned Research	1-8
1.9 Thesis Overview	1-8
 II. Subsystems, Reference Frames, and Kalman Filtering	 2-1
2.1 Important Subsystems in CIRIS, ARS, and NRS	2-1
2.1.1 Gimbaled INS	2-1

	Page
2.1.2 Strapdown INS	2-2
2.1.3 Baro-Altimeter and Central Air Data Computer . .	2-3
2.1.4 Range/Range-Rate Transponder System	2-3
2.1.5 Global Positioning System	2-4
2.2 Systems Descriptions	2-4
2.2.1 CIRIS	2-4
2.2.2 ARS	2-5
2.2.3 NRS	2-6
2.3 Reference Frames	2-6
2.4 Reference Frame Definitions	2-7
2.4.1 Reference Frame Transformation Matrices	2-10
2.5 WGS-84 Geodetic System	2-13
2.6 Kalman Filter Theory	2-14
2.6.1 Linear Kalman Filtering	2-15
2.6.2 Linearized and Extended Kalman Filtering	2-18
2.7 Summary	2-21
III. INS and Baro-Altimeter Models	3-1
3.1 Introduction	3-1
3.2 Litton LN-93 INS	3-1
3.2.1 Error Types	3-1
3.2.2 Error State Vector	3-4
3.3 Baro-Altimeter Model Revision	3-5
3.3.1 Litton Vertical Channel Aiding	3-5
3.3.2 Revised Baro-Altitude Model	3-7
3.4 Simulated Fighter Flight Profile	3-9
3.5 Summary	3-9

	Page
IV. Range/Range-Rate Transponder System	4-1
4.1 Introduction	4-1
4.2 RRS Range Measurements	4-2
4.3 Range Calculation From INS Data	4-4
4.4 RRS Error-State Model Equations	4-6
4.5 Predicted RRS Measurement Equation	4-8
4.6 Transformation of User Position Error	4-10
4.7 Kalman Filter Gain	4-14
4.8 Transponder Physical Locations	4-17
4.9 Summary	4-18
V. Global Positioning System Model and Space Vehicle Orbit Calculations	5-1
5.1 Introduction	5-1
5.2 GPS Range Measurements	5-2
5.3 GPS Pseudo-Range Calculation From INS Data	5-4
5.4 GPS Error-State Model Equations	5-10
5.5 Predicted GPS Measurement Equation	5-13
5.6 GPS Measurement Estimation	5-15
5.7 GPS Measurement Matrix	5-16
5.8 SV Measurement Set Selection	5-16
5.9 SV Initial Orbital Parameters	5-18
5.10 Summary	5-19
VI. Results	6-1
6.1 Monte Carlo Analysis of the NRS Error Model	6-1
6.2 Feedback and Platform Alignment	6-2
6.3 Performance Analysis Baseline	6-4
6.4 96-State Error Model Performance	6-7

	Page
6.5 72-State Error-Model	6-10
6.6 98-State Error-Model	6-12
6.7 128-State INS/RRS/GPS (NRS) Model	6-13
6.7.1 128-State NRS Model: GPS Measurements Only . .	6-14
6.8 128-State INS/RRS/GPS (NRS) Model	6-16
6.8.1 128-State NRS Model: GPS and RRS Measurements	6-16
6.9 Summary	6-19
 VII. Conclusions and Recommendations	 7-1
7.1 Baro-Altimeter Model	7-1
7.2 GPS Model	7-2
7.3 Truth Model	7-2
7.4 Monte Carlo Analyses	7-3
7.5 Opportunities for Future Research	7-4
 Appendix A. NRS Error Model State Definitions	 A-1
A.1 INS and Baro-altimeter Error States	A-1
A.2 RRS Transponder Error States	A-1
A.3 GPS Error States	A-1
 Appendix B. Litton LN-93 Error-State Model Dynamics Matrix	 B-1
B.1 Elements of the Process Noise Matrix	B-6
 Appendix C. 93-State INS Alignment and Flight Simulations	 C-1
C.1 93-State Model: Litton Alignment	C-2
C.2 93 State Model: Litton Fighter Flight	C-9
 Appendix D. 96-State INS Alignment and Flight Simulations	 D-1
D.1 96-State Model: Litton Alignment	D-1
D.2 96 State Model: Litton Fighter Flight	D-8

	Page
Appendix E. 96-State INS Alignment and Flight Simulations: Part II . .	E-1
E.1 96-State Model: Holloman Alignment	E-1
E.2 96-State Model: Holloman Fighter Flight	E-8
Appendix F. 72-State Reduced INS Model Validation	F-1
F.1 72-State Model: Holloman Alignment	F-2
Appendix G. 98-State INS-RRS Alignment and Flight Simulations	G-1
G.1 98-State Model: Holloman Alignment	G-1
G.2 98-State Model: Holloman Fighter Flight	G-10
Appendix H. 128-State INS-GPS Alignment and Flight Simulations . . .	H-1
H.1 128-State Model: Holloman Alignment	H-1
H.2 128-State Model: Holloman Fighter Flight	H-15
Appendix I. 128-State INS-RRS-GPS Alignment and Flight Simulations	I-1
I.1 128-State Model: Holloman Alignment	I-1
I.2 128-State Model: Holloman Fighter Flight	I-15
Bibliography	BIB-1
Vita	VITA-1

List of Figures

Figure	Page
2.1. Coordinate Frame Geometry	2-7
3.1. Basic Baro-Aided LN-93 INS	3-2
3.2. LN-93 Baro-Altimeter Error Model	3-6
3.3. LN-93 Vertical Channel Error Model	3-7
3.4. Revised Baro-Altimeter Error Model	3-8
3.5. Fighter Mission Flight Profile	3-10
4.1. Baro-Aided INS with RRS Transponder Aiding	4-2
4.2. Two Dimensional Earth-User-Transponder System	4-3
5.1. NRS: LN-93 INS Aided by Baro-Altimeter, RRS, and GPS	5-2
5.2. Orbital Path for a Typical SV	5-7
5.3. Planar Representation of SV Orbital Path	5-8
5.4. Orthogonal Orbital Frame Definition	5-9
5.5. Two-Dimensional Representation of User and SV Position Errors.	5-13
5.6. Optimal SV Set Geometry	5-17
6.1. 96-State Model, 2-Hour Flight: Latitude, Longitude Filter-Computed σ Plots	6-10
6.2. 98-State Model, 2-Hour Flight: Latitude, Longitude Filter-Computed σ Plots (RRS Measurements Only)	6-14
6.3. 128-State Model, 2-Hour Flight: Latitude, Longitude Filter-Computed σ Plots (GPS Measurements Only)	6-17
6.4. 128-State Model, 2-Hour Flight: Latitude, Longitude Filter-Computed σ Plots (RRS and GPS Measurements)	6-18
C.1. Alignment: 93-State Model (a) Latitude and (b) Longitude Errors	C-3

Figure	Page
C.2. Alignment: 93-State Model (a) East Tilt and (b) North Tilt Error States	C-4
C.3. Alignment: 93-State Model (a) Azimuth and (b) East Velocity Error States	C-5
C.4. Alignment: 93-State Model (a) North Velocity and (b) Vertical Velocity Error States	C-6
C.5. Alignment: 93-State Model (a) INS Altitude Error State and (b) Baro-Altimeter Total Error	C-7
C.6. Alignment: 93-State Model (a) ΔS_3 and (b) ΔS_4 Error States	C-8
C.7. Flight: 93-State Model (a) Latitude and (b) Longitude Error States . .	C-10
C.8. Flight: 93-State Model (a) East Tilt and (b) North Tilt Error States .	C-11
C.9. Flight: 93-State Model (a) Azimuth and (b) East Velocity Error States	C-12
C.10. Flight: 93-State Model (a) North Velocity and (b) Vertical Velocity Error States	C-13
C.11. Flight: 93-State Model (a) INS Altitude Error State and (b) Baro-Altimeter Total Error	C-14
C.12. Flight: 93-State Model (a) ΔS_3 and (b) ΔS_4 Error States	C-15
D.1. Alignment: 96-State Model (a) Latitude and (b) Longitude Errors . .	D-2
D.2. Alignment: 96-State Model (a) East Tilt and (b) North Tilt Error States	D-3
D.3. Alignment: 96-State Model (a) Azimuth and (b) East Velocity Error States	D-4
D.4. Alignment: 96-State Model (a) North Velocity and (b) Vertical Velocity Error States	D-5
D.5. Alignment: 96-State Model (a) INS Altitude Error State and (b) Baro-Altimeter Total Error	D-6
D.6. Alignment: 96-State Model (a) ΔS_3 and (b) ΔS_4 Error States	D-7
D.7. Flight: 96-State Model (a) Latitude and (b) Longitude Error States . .	D-9
D.8. Flight: 96-State Model (a) East Tilt and (b) North Tilt Error States .	D-10
D.9. Flight: 96-State Model (a) Azimuth and (b) East Velocity Error States	D-11

Figure	Page
D.10.Flight: 96-State Model (a) North Velocity and (b) Vertical Velocity Error States	D-12
D.11.Flight: 96-State Model (a) INS Altitude Error State and (b) Baro-Altimeter Total Error	D-13
D.12.Flight: 96-State Model (a) ΔS_3 and (b) ΔS_4 Error States	D-14
E.1. Alignment: 96-State Model (a) Latitude and (b) Longitude Errors . .	E-2
E.2. Alignment: 96-State Model (a) East Tilt and (b) North Tilt Error States	E-3
E.3. Alignment: 96-State Model (a) Azimuth and (b) East Velocity Error States	E-4
E.4. Alignment: 96-State Model (a) North Velocity and (b) Vertical Velocity Error States	E-5
E.5. Alignment: 96-State Model (a) INS Altitude Error State and (b) Baro-Altimeter Total Error	E-6
E.6. Alignment: 96-State Model (a) ΔS_3 and (b) ΔS_4 Error States	E-7
E.7. Flight: 96-State Model (a) Latitude and (b) Longitude Error States . .	E-9
E.8. Flight: 96-State Model (a) East Tilt and (b) North Tilt Error States .	E-10
E.9. Flight: 96-State Model (a) Azimuth and (b) East Velocity Error States	E-11
E.10.Flight: 96-State Model (a) North Velocity and (b) Vertical Velocity Error States	E-12
E.11.Flight: 96-State Model (a) INS Altitude Error State and (b) Baro-Altimeter Total Error	E-13
E.12.Flight: 96-State Model (a) ΔS_3 and (b) ΔS_4 Error States	E-14
F.1. Alignment: 96-State vs. 72-State Model (a) Latitude and (b) Longitude Errors	F-3
F.2. Alignment: 96-State vs. 72-State Model (a) East Tilt and (b) North Tilt Error States	F-4
F.3. Alignment: 96-State vs. 72-State Model (a) Azimuth and (b) East Velocity Error States	F-5

Figure	Page
F.4. Alignment: 96-State vs. 72-State Model (a) North Velocity and (b) Vertical Velocity Error States	F-6
F.5. Alignment: 96-State vs. 72-State Model (a) INS Altitude Error State and (b) Baro-Altimeter Total Error	F-7
G.1. Alignment: 98-State Model (a) Latitude and (b) Longitude Errors . .	G-2
G.2. Alignment: 98-State Model (a) East Tilt and (b) North Tilt Error States	G-3
G.3. Alignment: 98-State Model (a) Azimuth and (b) East Velocity Error States	G-4
G.4. Alignment: 98-State Model (a) North Velocity and (b) Vertical Velocity Error States	G-5
G.5. Alignment: 98-State Model (a) INS Altitude Error State and (b) Baro-Altimeter Total Error	G-6
G.6. Alignment: 98-State Model (a) ΔS_3 and (b) ΔS_4 Error States	G-7
G.7. Alignment: 98-State Model (a) Transponder 1, X Axis Position (b) Transponder 1, Y Axis Position Error States	G-8
G.8. Alignment: 98-State Model (a) Transponder 1, Z Axis Position (b) Transponder 1, Atmospheric Propagation Error States	G-9
G.9. Flight: 98-State Model (a) Latitude and (b) Longitude Error States . .	G-11
G.10. Flight: 98-State Model (a) East Tilt and (b) North Tilt Error States .	G-12
G.11. Flight: 98-State Model (a) Azimuth and (b) East Velocity Error States	G-13
G.12. Flight: 98-State Model (a) North Velocity and (b) Vertical Velocity Error States	G-14
G.13. Flight: 98-State Model (a) INS Altitude Error State and (b) Baro-Altimeter Total Error	G-15
G.14. Flight: 98-State Model (a) ΔS_3 and (b) ΔS_4 Error States	G-16
G.15. Flight: 98-State Model (a) Transponder 1, X Axis Position (b) Transponder 1, Y Axis Position Error States	G-17
G.16. Flight: 98-State Model (a) Transponder 1, Z Axis Position (b) Transponder 1, Atmospheric Propagation Error States	G-18

Figure	Page
H.1. Alignment: 128-State Model (a) Latitude and (b) Longitude Errors . .	H-3
H.2. Alignment: 128-State Model (a) East Tilt and (b) North Tilt Error States	H-4
H.3. Alignment: 128-State Model (a) Azimuth and (b) East Velocity Error States	H-5
H.4. Alignment: 128-State Model (a) North Velocity and (b) Vertical Velocity Error States	H-6
H.5. Alignment: 128-State Model (a) INS Altitude Error State and (b) Baro-Altimeter Total Error	H-7
H.6. Alignment: 128-State Model (a) ΔS_3 and (b) ΔS_4 Error States	H-8
H.7. Alignment: 128-State Model (a) Transponder 1, X Axis Position (b) Transponder 1, Y Axis Position Error States	H-9
H.8. Alignment: 128-State Model (a) Transponder 1, Z Axis Position (b) Transponder 1, Atmospheric Propagation Error States	H-10
H.9. Alignment: 128-State Model (a) User Clk Bias and (b) User Clock Drift States (Note: GPS Measurements Only)	H-11
H.10. Alignment: 128-State Model: Steady State (a) User Clk Bias and (b) User Clock Drift States (Note: GPS Measurements Only)	H-12
H.11. Alignment: 128-State Model (a) SV 1, Codeloop and (b) SV 1, Atmospheric Propagation Error States (Note: GPS Measurements Only) . . .	H-13
H.12. Alignment: 128-State Model (a) SV 1, Clock and (b) SV 1, X Position Error States (Note: GPS Measurements Only)	H-14
H.13. Flight: 128-State Model (a) Latitude and (b) Longitude Error States .	H-16
H.14. Flight: 128-State Model (a) East Tilt and (b) North Tilt Error States .	H-17
H.15. Flight: 128-State Model (a) Azimuth and (b) East Velocity Error States	H-18
H.16. Flight: 128-State Model (a) North Velocity and (b) Vertical Velocity Error States	H-19
H.17. Flight: 128-State Model (a) INS Altitude Error State and (b) Baro-Altimeter Total Error	H-20
H.18. Flight: 128-State Model (a) ΔS_3 and (b) ΔS_4 Error States	H-21

Figure	Page
H.19. Flight: 128-State Model (a) Transponder 1, X Axis Position (b) Transponder 1, Y Axis Position Error States	H-22
H.20. Flight: 128-State Model (a) Transponder 1, Z Axis Position (b) Transponder 1, Atmospheric Propagation Error States	H-23
H.21. Flight: 128-State Model (a) User Clk Bias and (b) User Clock Drift States (Note: GPS Measurements Only)	H-24
H.22. Flight: 128-State Model (a) SV 1, Code loop and (b) SV 1, Atmospheric Propagation Error States (Note: GPS Measurements Only)	H-25
H.23. Flight: 128-State Model (a) SV 1, Clock and (b) SV 1, X Position Error States (Note: GPS Measurements Only)	H-26
I.1. Alignment: 128-State Model (a) Latitude and (b) Longitude Errors . .	I-3
I.2. Alignment: 128-State Model (a) East Tilt and (b) North Tilt Error States	I-4
I.3. Alignment: 128-State Model (a) Azimuth and (b) East Velocity Error States	I-5
I.4. Alignment: 128-State Model (a) North Velocity and (b) Vertical Velocity Error States	I-6
I.5. Alignment: 128-State Model (a) INS Altitude Error State and (b) Baro-Altimeter Total Error	I-7
I.6. Alignment: 128-State Model (a) ΔS_3 and (b) ΔS_4 Error States	I-8
I.7. Alignment: 128-State Model (a) Transponder 1, X Axis Position (b) Transponder 1, Y Axis Position Error States	I-9
I.8. Alignment: 128-State Model (a) Transponder 1, Z Axis Position (b) Transponder 1, Atmospheric Propagation Error States	I-10
I.9. Alignment: 128-State Model (a) User Clk Bias and (b) User Clock Drift States (Note: Both GPS Measurements Only)	I-11
I.10. Alignment: 128-State Model: <i>Steady State</i> (a) User Clk Bias and (b) User Clock Drift States (Note: GPS and RRS Measurements)	I-12
I.11. Alignment: 128-State Model (a) SV 1, Code loop and (b) SV 1, Atmospheric Propagation Error States (Note: Both GPS Measurements Only)	I-13

Figure	Page
I.12. Alignment: 128-State Model (a) SV 1, Clock and (b) SV 1, X Position Error States (Note: Both GPS Measurements Only)	I-14
I.13. Flight: 128-State Model (a) Latitude and (b) Longitude Error States .	I-16
I.14. Flight: 128-State Model (a) East Tilt and (b) North Tilt Error States .	I-17
I.15. Flight: 128-State Model (a) Azimuth and (b) East Velocity Error States	I-18
I.16. Flight: 128-State Model (a) North Velocity and (b) Vertical Velocity Error States	I-19
I.17. Flight: 128-State Model (a) INS Altitude Error State and (b) Baro-Altimeter Total Error	I-20
I.18. Flight: 128-State Model (a) ΔS_3 and (b) ΔS_4 Error States	I-21
I.19. Flight: 128-State Model (a) Transponder 1, X Axis Position (b) Transponder 1, Y Axis Position Error States	I-22
I.20. Flight: 128-State Model (a) Transponder 1, Z Axis Position (b) Transponder 1, Atmospheric Propagation Error States	I-23
I.21. Alignment: 128-State Model (a) User Clk Bias and (b) User Clock Drift States (Note: RRS and Both GPS Measurements)	I-24
I.22. Alignment: 128-State Model (a) SV 1, Codeloop and (b) SV 1, Atmospheric Propagation Error States (Note: RRS and Both GPS Measurements)	I-25
I.23. Alignment: 128-State Model (a) SV 1, Clock and (b) SV 1, X Position Error States (Note: RRS and Both GPS Measurements)	I-26

List of Tables

Table	Page
2.1. WGS 84 Parameters	2-14
4.1. RRS Transponder Locations	4-17
5.1. GPS Optimal Constellation Initial Conditions	5-19
A.1. NRS System Model: INS States 1 \rightarrow 22	A-2
A.2. NRS System Model: INS States 23 \rightarrow 50	A-3
A.3. NRS System Model: INS States 51 \rightarrow 72	A-4
A.4. NRS System Model: INS States 73 \rightarrow 96	A-5
A.5. NRS System Model: RRS Error States	A-6
A.6. NRS System Model: GPS Error States	A-7
B.1. Elements of the Dynamics Submatrix F_{11}	B-2
B.2. Elements of the Dynamics Submatrix F_{12}	B-3
B.3. Elements of the Dynamics Submatrix F_{13}	B-3
B.4. Elements of the Dynamics Submatrix F_{14}	B-4
B.5. Elements of the Dynamics Submatrix F_{15}	B-4
B.6. Elements of the Dynamics Submatrix F_{16}	B-5
B.7. Elements of the Dynamics Submatrix F_{22}	B-5
B.8. Elements of the Dynamics Submatrix F_{33}	B-5
B.9. Non-zero Elements of Process Noise Submatrix Q_{11}	B-6
B.10. Non-zero Elements of Process Noise Submatrix Q_{22}	B-6

Abstract

To quantify the performance abilities of existing or proposed navigation systems, the U.S. Air Force has for the last several years compared the performance of the system under test to the performance of a *baseline* navigation system known as the Completely Integrated Reference Instrumentation System (CIRIS). CIRIS obtains a highly accurate navigation solution by combining the output from three major subsystems: inertial navigation system (INS) information, barometric altitude information, and range and range-rate data from ground transponders which have been precisely surveyed. Although the navigation solution produced by CIRIS is highly accurate, it will soon be inadequate as the standard against which future navigation systems can be tested. This research proposes an alternative to CIRIS - a hybrid Navigation Reference System (NRS) which is designed to take advantage of a newer INS (the LN-93), certain features of the current CIRIS, and certain features of the Global Positioning System (GPS). Analysis is conducted using a Kalman filter development package known as the Multimodel Simulation for Optimal Filter Evaluation (MSOFE). Both a large order *truth* model for the NRS (in which a full 24 satellite constellation is modeled) and a full-order Kalman filter are developed. Results suggest that the proposed NRS (with GPS aiding) provides a significantly improved navigation solution as compared to CIRIS.

A NAVIGATION REFERENCE SYSTEM (NRS) USING GLOBAL POSITIONING SYSTEM (GPS) AND TRANSPONDER AIDING

I. Introduction

This thesis constitutes an extension of previous work in which the goal has been to design, implement, and analyze an improved Kalman filter for the Completely Integrated Reference Instrumentation System (CIRIS). Developed at the Central Inertial Guidance Test Facility (CIGTF), Holloman AFB, NM, CIRIS functions as a high fidelity navigation system *standard or reference* against which various INS mechanizations are tested.

1.1 Background

In order to quantify the accuracy of existing or proposed navigation systems, it is necessary to compare the navigation solution of the system under test to the solution produced by a *reference system* such as CIRIS [17, 18]. Additionally, the *reference system* must be at least an order of magnitude more accurate when compared to the navigation systems which are currently tested at CIGTF, as well as those which will be tested there in the near future [17, 28]. CIRIS has served as the high quality navigation system test standard since becoming operational in 1975. However, due to expected improvements in future navigation systems, CIRIS may no longer be adequate as a navigation system standard. Consequently, incorporation of Global Positioning System (GPS) aiding to improve the CIRIS navigation solution is being pursued. It is believed that GPS aiding will improve the navigation solution accuracy well beyond the accuracy which is obtained by the current implementation of CIRIS [18, 32, 33].

Plans exist at CIGTF to implement an upgraded system (similar to CIRIS) which is called the Advanced Reference System (ARS). The ARS is designed to function in much the same way that CIRIS currently does, but is implemented with a state-of-the-art inertial navigation system (INS) and is aided by GPS. One candidate INS being considered for use

in the ARS is the "strap-down" Litton LN-93 [17]. The accuracy and reliability of the ring laser gyro (RLG) inertial systems, such as the LN-93, make them better candidates for use in ARS than the older LN-39 gimbaled INS which is used in CIRIS. The newer INS, along with the addition of GPS aiding, will ensure that ARS is a more reliable test *standard* against which future navigation systems may be tested. However, the performance specifications of ARS remain unchanged with the exception of some improvement in the vertical channel.

1.2 Problem Statement

In order to accommodate future INS testing requirements, a reference system of much higher accuracy is required. Such accuracy may be obtained with a properly integrated system including INS, GPS, and existing transponders. A fundamental first step in this process is the development of a *truth model* and a full-order extended Kalman filter which incorporates the LN-93 INS system model, transponder aiding (as implemented in CIRIS), and GPS aiding, must be designed, implemented, and analyzed [17, 18]. The truth model proposed in this thesis is called the Navigation Reference System (NRS).

1.3 Summary of Applicable Previous Research

Although a substantial body of knowledge exists with respect to the GPS-INS integration problem, published research which relates specifically to GPS-CIRIS or GPS-ARS integration is limited. Only Solomon's research *specifically* relates to GPS-CIRIS integration [32, 33]. Other research which is relevant to this thesis includes the CIRIS research performed by Snodgrass [31] and a group project completed by students in the AFIT navigation sequence in which the LN-93 INS error model is used in an INS-GPS integration problem [11].

1.3.1 GPS-INS Integration. Cunningham addressed the problem of the "filter-driving-filter" instability which can occur when two systems (INS and GPS, in this case) are used to aid one another while each of them employs its own optimal Kalman filter [7, 8]. Cunningham assumed a local-level (north, east, down) INS platform using three gyros and

three accelerometers. The GPS receiver was a four-channel set capable of simultaneously obtaining range measurements from four GPS satellites (also called space vehicles or SVs). *GPS range-rate measurements were not modeled due to the assumption of the high-dynamic conditions (which would preclude an actual GPS receiver from accurately calculating range-rate).* Cunningham developed truth models for both the INS and GPS. Kalman filters based on the truth models were combined to produce a joint-solution Kalman filter which was believed to be the best possible structure for such an integration [7, 8].

Cunningham concluded that a combined INS-GPS mechanization provides the user with the high accuracy of the GPS during benign conditions, and provides the beneficial characteristics of the INS during high dynamic maneuvers [7]. His conclusion is consistent with other authors who discuss the INS-GPS integration problem [6, 22].

1.3.2 GPS-CIRIS Integration. In addition to being directly related to the GPS-CIRIS integration problem, Solomon's work represents a significant extension of Cunningham's work [7, 8] because the GPS error model is improved considerably and the CIRIS extended Kalman filter is redesigned to enhance CIRIS performance [32, 33]. Solomon's preliminary GPS error model [32] (and extensions developed in research conducted as part of the navigation systems class project referenced previously [11]) forms the basis for the GPS error model used in this thesis.

The thrust of Solomon's work was to produce an improved truth model for CIRIS, to produce a reduced-order Kalman filter for CIRIS, and to integrate GPS aiding for the CIRIS navigation solution. In his thesis [33], he completed a 127-state CIRIS truth model. He also produced a reduced-order Kalman filter which had 70 states. In a special study [32], Solomon also assembled a (stationary SV) GPS error model and implemented it in software.

Solomon concluded that the 70-state Kalman filter emulated the 127-state truth model very well. (Performance indicators for the 70-state filter appear to be slightly better than a 127-state Kalman filter which was based on the *truth model*. This may be attributable to lack of observability of certain states in the truth model [17, 25].)

1.3.3 CIRIS Kalman Filter Improvements. Solomon made it clear in his conclusions that he believed validation of the models he had developed was more important than simply tuning the filters. As a result, Snodgrass had as his thesis objectives to improve the structure and efficiency of the filter software, and to validate a *refined* version of the 70-state Kalman filter originally developed by Solomon. Snodgrass achieved the first objective after considerable software development. The second objective was satisfied by comparing the refined 70-state Kalman filter performance to *truth data* in the form of position and velocity data collected on the CIGTF test track. Additional comparisons were made between the improved Kalman filter and position data obtained from the laser ranging system at the Yuma test range. Due to the amount of time needed to accomplish the primary objectives discussed above, Snodgrass did not work specifically on the GPS integration problem [30].

Snodgrass concluded that the 70-state Kalman filter performed better than the older version of the CIRIS filter in most cases. The exception is in the case of range-rate measurement processing. The new filter range-rate residuals were unacceptably large at certain times in the flight, resulting in a large percentage of the measurements being rejected. Snodgrass concludes with a recommendation that GPS integration (within a full order truth model) be the next step at AFIT.

1.4 Research Objectives

References cited in the previous section [32, 33, 31, 11] form the foundation upon which much of this research is based. The major objectives of this thesis are essentially to consolidate and *extend* previous work as follows:

1. Revise the barometric altimeter error model to improve the fidelity of the vertical channel aiding model contained in the LN-93 documentation and compare the performance of the new model to the old model.
2. Write software to calculate satellite positions of a 24-GPS-SV constellation for a two hour period.
3. Assemble a high fidelity truth model (NRS) which integrates GPS aiding.

4. Produce a *full-order* Kalman filter based on the truth model.
5. Conduct Monte Carlo analyses to determine performance characteristics of the configurations shown below.
 - (a) Evaluate performance of the original 93-state INS error model for an alignment and for a flight profile. Initial conditions are identical to those used in the Litton validation work.
 - (b) Evaluate performance of the revised (new baro model) 96-state INS error model for an alignment and for a flight profile. Initial conditions are identical to those used in the Litton validation work.
 - (c) Evaluate performance of the revised 96-state INS error model for an alignment and for a flight profile. Initial conditions are those for Holloman AFB, NM.
 - (d) Evaluate performance of a reduced 72-state INS error model for an alignment and for a flight profile. Initial conditions are those for Holloman AFB, NM.
 - (e) Evaluate performance of a 98-state reduced INS plus ground transponder error model for an alignment and for a flight profile. Initial conditions are those for Holloman AFB, NM.
 - (f) Evaluate performance of the 128-state NRS error model (which integrates INS, RRS, and GPS error models) for an alignment and for a flight profile. Initial conditions are those for Holloman AFB, NM.

1.5 Assumptions

The problem described in the problem statement requires some assumptions. First, the LN-93 error model is accepted as the *truth model* for the INS subsystem of the NRS, with the notable exception of the baro-altitude portion of that model. The baro-altitude model is replaced with a revised model as discussed in Chapter III. Although there is some verification that results comparable to those obtained by Litton are achieved in this research, exhaustive INS error model validation is not undertaken.

The error model for the ground transponder range/range-rate system (RRS) from the Snodgrass thesis [31] is used as the error model for the RRS subsystem of the NRS. RRS initial conditions and tuning parameters are also extracted from [31].

A good error model for the GPS error sources is extracted from separate articles by Cox [6], Martin [21], and Milliken and Zoller [27] and presented in [11]. The error states in the model presented in [11] are used as the basis for the GPS subsystem in the NRS model, although different initial conditions and tuning parameters based on personal experience are used in some cases to improve performance of the Kalman filter.

1.6 Research Approach

This section provides a general outline of the approach used to accomplish each of the tasks listed in the research objectives section. The correspondence between items in this section and those in Section 1.4 are NOT one-to-one; some tasks overlap with one another.

1. Revision of the barometric altimeter error model is necessary because improving model fidelity will enhance the system (*truth*) model [17, 31]. The revised model is based on the Litton LN-93 error model documentation and discussions with Lewantowicz [17].
2. A *truth model* and full-order *benchmark* Kalman filter are constructed. The truth model is assembled as follows:
 - (a) The Litton LN-93 documentation is the basis for the INS error model [20]. The single barometric altimeter state in the LN-93 model is replaced by the revised model discussed above. Baro-altimeter error dynamic equations are developed in Chapter III.
 - (b) The RRS transponder portion of the NRS error model is based on the work performed by Snodgrass [31].
 - (c) A revised GPS error model is constructed for the NRS. *GPS SV motion is modeled in this effort whereas stationary SVs were assumed previously* [11, 33].

Software which explicitly calculates time-varying SV position for a full GPS constellation is written.

3. Analyses are conducted using the Multimode Simulation for Optimal Filter Evaluation (MSOFE) software package [5]. MSOFE is run in a variety of modes and configurations in order to establish a performance baseline for the (INS-GPS) *truth model*. This objective is important because there has been no such analysis performed on the GPS truth model to date. This analysis also provides a baseline against which full-order and reduced-order filters may be compared.

1.7 Computer and Software Usage

Simulations performed as part of this research require a substantial amount of computing power. This need stems from the calculation-intensive nature of time domain simulations of dynamic systems in general and the size of the NRS integration problem in particular. Both the truth model and the full-order Kalman filter each have 128 states. The filter covariance matrix is treated as an upper diagonal array in software, yet still contains 8,256 elements (each of which must be integrated continuously). A "dedicated" VAX-station III (operating at approximately 3 MIPS) is inadequate for the task. Turn around times for the flight simulations are on the order of several days. A tremendous improvement is achieved by hosting and running the NRS simulation on a MicroVax III within which a 15 MIPS coprocessing board is installed. However, the problem is still nontrivial. More than 18 hours are needed to complete a 10-run Monte Carlo alignment simulation with 128 states in the truth model and 128 states in the filter. A 10-run Monte Carlo flight simulation (presented in Chapter III) is expected to take 8 to 9 days.

In addition to the software which is written explicitly for this thesis, three software packages of a more general nature are used to support the research conducted. MSOFE (the Multi-mode Simulation for Optimal Filter Evaluation), written by Dr Neal Carlson and Mr Stan Musick, functions as a "shell" within which proposed or existing Kalman filter designs may be implemented and tested [5]. PROFGEN (short for PROFile GENERator), also written by Musick, is used to create time histories of variables such as aircraft position

in R^3 space, attitude, and acceleration [1]. The time histories created in this manner serve as the "truth" (or nominal) aircraft trajectory in MSOFE simulations of the type conducted in this thesis. PROFGEN is used to create the aircraft trajectory in the two-hour flight simulation which is discussed in more detail in Chapter III. Data which are produced during the MSOFE "runs" are post-processed and plotted using *MATRIX_N* [14]. Analyses of several barometric altitude models and some observability analyses are also performed using *MATRIX_N*.

1.8 Summary of Planned Research

While this thesis is intended to consolidate and extend the work begun by Cunningham, Solomon, and Snodgrass [7, 32, 33, 31], it departs from previous efforts somewhat. CIRIS accuracy as a reference is expected to be inadequate in the very near future, if it is not already. CIGTF is in the process of designing a new system such as ARS which provides performance comparable to CIRIS, but which records INS, GPS, and transponder data which can be post-processed with high-accuracy algorithms. Therefore, the primary goal of this thesis is to design, implement, and test a system truth model and a benchmark Kalman filter which is called the the Navigation Reference System (NRS). To accomplish the thesis objectives, a system truth model is assembled, implemented, and analyzed using the MSOFE software package.

1.9 Thesis Overview

Chapter II provides brief descriptions of the subsystems which compose the CIRIS, ARS, and NRS. Reference frames and general Kalman filter theory are also presented. Chapter III discusses the relevant INS theory. The LN-93 error model is presented and its role in the NRS is discussed. The revised baro-altimeter model is developed and discussed. A full development of the RRS error and measurement models is presented in Chapter IV. A similar development for GPS is included in Chapter V, along with the development of GPS SV position calculations.

Chapter VI presents the results of simulations and analyses for both the truth model and the benchmark Kalman filter. Five configurations are evaluated and compared. Chap-

ter VII summarizes the thesis effort. Conclusions are drawn from the results presented in Chapter VI and recommendations are made for future research efforts.

Appendix A includes the *complete* 152 error-state vector in a set of tables which are arranged to emphasize the natural divisions in the model. Appendix B is obtained from the Litton reference [20] and shows the non-zero elements in the various sub-blocks in the LN-93 dynamics matrix F , and the non-zero elements of the noise process matrix Q , which is also extracted from the Litton reference [20].

Appendix C includes the data plots from the "baseline" case of an INS operating with baro-aiding only. Two cases are shown: first, the results from a simulated 8-minute alignment are presented, and second, the results from a two-hour fighter flight simulation (which is explained in greater detail in Chapter III). These simulations are initialized with parameters which were used by Litton in establishing the baseline performance of the LN-93 INS [20]. The initial latitude is 45 degrees north, and longitude is zero degrees. Initial altitude is zero feet. Appendix D includes plots for the 96-state INS (which includes a new baro-altimeter model). The same two cases are shown (i.e. alignment and flight runs). The same initial latitude, longitude, and altitude as discussed above are used in this case. Appendix E presents plots for the 96-state INS (with the new baro-altimeter model). Both the alignment and flight runs are repeated, but with initial latitude, longitude, and altitude parameters set to the appropriate values for Holloman AFB, NM.

In Appendix F, results from a reduced INS error model are presented. The INS error model is reduced to 72 states. The 72-state truth model error behavior is compared to that of the 96-state truth model.

Appendix G includes plots for the 98-state (reduced) INS and RRS configuration. The same two cases are shown (i.e. alignment and flight runs), again with initial conditions set for Holloman AFB. Appendices H and I each contain plots for the 128-state INS/RRS and GPS configuration. In both cases, the alignment and flight run results are shown, and initial conditions are those for Holloman AFB. Appendix H presents results from operating NRS with GPS aiding only, whereas Appendix I demonstrates NRS performance with measurements from both the RRS and GPS subsystems.

II. Subsystems, Reference Frames, and Kalman Filtering

2.1 Important Subsystems in CIRIS, ARS, and NRS

Several concepts related to CIRIS and ARS operation are central to understanding the purpose and need for this thesis research. Therefore, it is useful to begin with a brief description of the subsystems which make up CIRIS and ARS. NRS subsystems will also be described. The subsystems which are defined include both gimbaled and strapdown INS mechanizations, the barometric altimeter and central air data computer (CADC), the radio frequency range/range-rate transponder system, and the Global Positioning System (GPS). The details provided in this thesis for each of these subsystems are only at that level required for clear understanding of the subsystem function in the CIRIS, ARS, or NRS "integrated" systems. It is not within the scope of this thesis to provide extensive operational or functional details for these subsystems.

2.1.1 Gimbaled INS. Britting describes inertial navigation as the process of determining the position and velocity of a vehicle (such as an aircraft, ship, or space vehicle) with respect to a specified frame of reference by the use of calibrated electro-mechanical devices installed within the vehicle [3]. A typical inertial navigation system (INS) makes use of the output of gyroscopes and accelerometers to determine position and velocity precisely. Specifically, inertial systems perform the following functions:

1. instrument a known reference frame,
2. measure specific force and extract accelerations in the reference frame,
3. perform integrations of accelerations to obtain velocity and position.

A gyroscope (usually referred to as a gyro) is a device which *traditionally* employs a rapidly spinning mass to create a strong angular momentum vector. Since changes in the angular momentum of the gyro are proportional to applied torque, the gyro can maintain a known spatial orientation if appropriate torque control is applied. Gyros are typically used as sensing elements in closed-loop servo systems which operate to maintain the gyros' spatial orientation. Thus, spinning mass gyros have traditionally been the devices of choice

to instrument the three dimensional Cartesian coordinate frame which is referred to as the *inertial frame* in aircraft navigation [3]. However, some navigation systems use a more recently developed inertial instrument known as the ring laser gyroscope (RLG) [20].

RLG construction *typically* consists primarily of an optical cavity, a laser device, three mirrors, a prism, and a pair of photodetectors [29]. According to Savage, the RLG operates as follows. Two beams of (laser) light are propagated (opposing one another) around the path enclosed by the optical cavity [29]. If the cavity is rotating in an inertial sense, the propagation times of the two light beams is different. The delay manifests itself in the form of a phase shift between the two beams, and the phase shift is detected by a pair of photodetectors [29]. The magnitude of the phase shift provides a direct indication of the speed of angular displacement of the instrument [29]. Devices of this type are extremely reliable due to the absence of moving parts [29], and with recent developments in production techniques, are capable of accuracy approaching that of their gimbaled counterparts [20].

Specific force is measured by *accelerometers*. The most common accelerometers to date have been devices which are sophisticated variations of the simple pendulum [3]. The motion of the internal mass is related to the inertially referenced motion of the instrument (case) by Newton's second law of motion. However, to obtain the *correct* measure of inertial acceleration, the effects of local gravity must be removed from the measured specific force [3, 18]. Consequently, the navigation system must be capable of distinguishing between the local gravity field and accelerations which are being applied to the accelerometer as a result of flight dynamics. Otherwise, significant errors will appear in the velocity and position calculations and these errors will be compounded in subsequent system updates [3, 29].

The gyros and accelerometers described above are integrated into a single system called the INS. An example of a widely used *gimbaled* INS (as described above) is the Litton LN-39. The LN-39 role in CIRIS will be discussed in a later section.

2.1.2 Strapdown INS. The strapdown INS (also referred to as a strapped-down INS in the literature [29]) is conceptually quite similar to the gimbaled INS referenced in the previous section. The "strapdown" system obtains its name from the fact that

there are no platform gimbals in the system. For example, the LN-93 is mechanized by installing three single-degree-of-freedom (SDOF) ring laser gyros (RLGs) and three SDOF accelerometers to a rigid structure (in the case of the LN-93, a rigid platform within the INS "black box"). An internal computer maintains a current computation of aircraft attitude based on measurements obtained from the six inertial sensors. Although strapdown systems were previously less accurate than gimbaled INS mechanizations [3], they are becoming far more accurate and have already surpassed the precision achieved by some of their gimbaled counterparts [18]. Because it uses high accuracy RLGs, the LN-93 is a prime example of the state-of-the-art in strapdown INS technology. It is for this reason that the LN-93 is the INS selected for use in the NRS.

2.1.3 Baro-Altitude and Central Air Data Computer (CADC). A well documented shortcoming of any INS is the instability which (in the absence of aiding information) results in unbounded error growth in the vertical position and vertical velocity channels [3, 10, 16]. This inherent instability is controlled by *vertical channel aiding*. Such aiding is frequently accomplished with vertical position information provided from either a barometric altimeter or a CADC. Either of these two approaches provides vertical channel stability by providing additional measurements in the form of pressure altitude information. This *external* altitude information has the effect of stabilizing the vertical channels which would otherwise be divergent if position were calculated solely on the basis of information (measurements) obtained from inertial instruments [3]. The baro-altitude role in the NRS will be discussed in more detail in Chapter III.

2.1.4 Range/Range-Rate Transponder System. The range/range-rate transponder system (RRS) is designed specifically to augment the CIRIS INS in order to improve the CIRIS navigation solution accuracy. The RRS system consists of an interrogator unit which is carried on board the aircraft which carries CIRIS, and approximately forty (40) fixed-site ground transponders whose positions have been precisely surveyed [28]. The interrogator transmits moderate rate digital interrogation codes which are received by any of the ground transponders that are within broadcast range. If the transponder recognizes its own identifier, it will respond with a return signal. Range measurements are obtained by

comparing the phase of the outgoing (interrogator) signal with the incoming (transponder) signal. Because any electromagnetic signal experiences delays when propagated through a medium, the delay which is measured by the phase shift comparison provides a direct indication of slant range from the CIRIS aircraft to the transponder being interrogated [28]. Note that the RRS transponder system is also a source of vertical information which aids in improving the overall navigation solution of CIRIS, ARS, or NRS. The RRS transponder subsystem is discussed in detail in Chapter IV.

2.1.5 Global Positioning System (GPS). GPS is a navigation system which is based on world-wide coverage of a constellation of 24 space vehicles (called SVs), or satellites [12]. According to Milliken and Zoller [27], the GPS provides position and velocity information to users located anywhere in the world, with anticipated accuracy on the order of 10 meters. Normally, the user needs to acquire and maintain lock on four SVs in order to obtain a complete navigation solution. However, Milliken and Zoller also state that fewer SVs are needed if the user has other information (such as baro-altimeter measurements) available for supplementing the GPS information. The GPS navigation solution is obtained in a manner similar to the RRS transponder system described previously.

The GPS *pseudo-range* between the user and each SV is computed based on knowledge of time (the master GPS clock) and the unique signal format which is broadcast by each SV. Once the four ranges are known, a recursive algorithm is solved to compute the user's position [27]. The GPS subsystem model is presented in Chapter V.

2.2 Systems Descriptions

Subsystems which play crucial roles in the CIRIS, ARS, and NRS systems are described in previous sections. The complete systems are described below.

2.2.1 CIRIS. Since 1965, the 6585th Test Group at Holloman Air Force Base has conducted component and system testing to analyze and verify performance of state-of-the-art inertial navigation systems. Various systems have been used as the *standard* against which systems under test were compared. The most recent of the systems to be

called the test standard, CIRIS has been used with great success since 1975. Using a Kalman filter, CIRIS optimally combines data from several subsystems to obtain a highly accurate navigation solution which has (until recently) met or exceeded all requirements for navigation reference accuracy. CIRIS is configured to be carried aboard either cargo or fighter aircraft. Additionally, CIRIS may be operated in a mobile test van. This mode is used for low dynamic tests in order to save costs associated with flight testing.

In its current configuration, CIRIS combines INS, CADC, radio frequency RRS transponder system information, and a time referencing system which is connected to the Holloman Inter-Range Instrumentation Group (IRIG) time-keeping system. Currently, two versions of CIRIS are available. The first (CIRIS I) uses the Litton LN-15 INS, and the second (CIRIS II) uses the Litton LN-39 INS. Additionally, the CIRIS II system uses an updated CADC which is the same type as those used on the F-16 aircraft. (CIRIS II is not fully operational at this time due to difficulty with data collection software [28].)

The CIRIS navigation solution has been shown to be accurate to within 14 feet in the horizontal direction, and 40 feet in the vertical direction. However, CIRIS accuracy will soon be eclipsed by new navigation systems under development. The new navigation systems will take advantage of improved gyro and accelerometer technologies as well as GPS aiding [17, 18]. When such systems are produced, they will rival or surpass CIRIS accuracy. Although such accuracy in navigation systems *is in fact desirable*, there is no reliable means of testing the new systems unless a superior test *standard* can be devised. Thus, the need for the ARS (and/or NRS) is established.

2.2.2 ARS. The Advanced Reference System (ARS) is under development at Holloman AFB and is intended to provide a more reliable reference system than CIRIS, even though comparable performance capabilities are expected [17]. When ARS becomes operational, performance of future navigation systems may be validated by comparing them to the (post-processed) solution from ARS, in much the same way that current systems are compared to CIRIS I or II.

ARS will also employ a Kalman filter which optimally combines data from several subsystems including an INS (type undetermined), a CADC (type undetermined), and

the RRS which is used by CIRIS I and II. ARS will also make use of GPS pseudo range information which is now available. Because of the higher fidelity components and the integration of GPS aiding, ARS accuracy should improve significantly over that of CIRIS [28]. ARS is being packaged in an AIM-9 missile pod in order to facilitate external carriage of the reference system. This form of packaging allows quick installation and removal of the ARS reference package on aircraft which are used for flight testing of navigation systems.

2.2.3 NRS. The Navigation Reference System (NRS) includes a Kalman filter which combines subsystems including an INS (with a baro-altimeter model), the RRS used by CIRIS, and GPS. The NRS uses the LN-93 INS and a revised baro-altimeter model as well. As a result, NRS should be a viable test standard for navigation system testing, and in particular, may possibly be used as a "post-processing" Kalman filter and smoother [17] for the ARS. (Such use is contingent upon which subsystems are chosen for ARS. If incompatible subsystems are used, the NRS model is likely to be unsuitable for ARS applications.)

2.3 Reference Frames

A navigation "solution" has significance only if the coordinate frame in which the solution is expressed is clearly understood. While the preceding statement may seem obvious, it cannot be overemphasized. Consider that the Litton LN-93 documentation defines the earth frame, the true frame, the computer frame, the platform frame, the sensor frame, the gyro frame, the accelerometer frame, and the body frame [20].

Hence, there is an urgent need for precise notation which aids in the communication process. There is also a need for the ability to transform quantities which are expressed in one frame into appropriate expressions in another frame. This need gives rise to *transformation matrices*.

Solutions to both of the needs discussed above are provided by Britting [3]. The notation is introduced as needed. Prior to defining the transformation matrices, several reference frame definitions are needed. Figure (2.3) depicts the central reference frames which Britting has chosen to define. They are discussed in the next section.

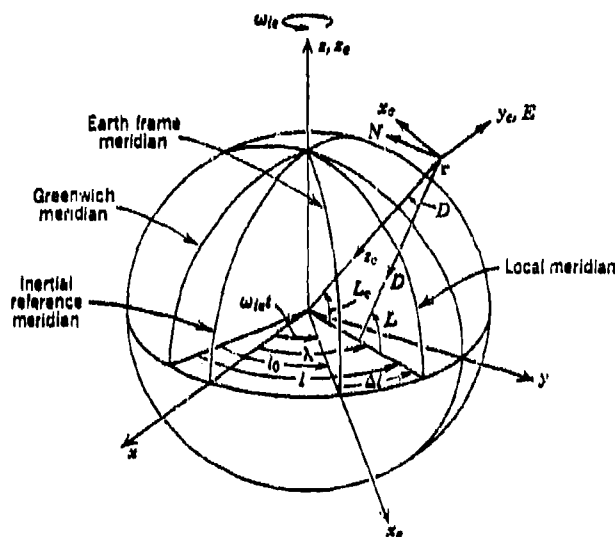


Figure 2.1. Coordinate Frame Geometry [3]

2.4 Reference Frame Definitions

The coordinate frame definitions below are those which are significant to this thesis [3]. Figure (2.3) depicts the first three frames which are defined. Figure (2.3) uses the following notation:

λ	\triangleq	Longitude with respect to inertial frame
Δl	\triangleq	Longitude with respect to ECEF frame
l_0	\triangleq	Initial longitudinal displacement of ECEF from Greenwich meridian (usually defined as ZERO)
L	\triangleq	Geographic frame latitude
L_c	\triangleq	Geocentric frame latitude
ω_{ie}	\triangleq	Earth angular rate (rad/sec)

1. Inertial frame: an orthogonal R^3 coordinate system; its origin is coincident with the earth's center of mass and the frame is oriented as follows. The x_i, y_i plane lies in the earth's equatorial plane and does not rotate with respect to the fixed stars. The

z_i axis always points toward Aries. The z_i axis projects from the earth's center of mass directly through the North pole. (This frame is depicted by the $[x, y, z]$ frame in Figure (2.3.)

2. Earth-Centered-Earth-Fixed (ECEF) frame: an orthogonal R^3 coordinate system; its origin is coincident with the earth's center of mass. The x_e, y_e plane lies in the earth's equatorial plane and the x_e axis is *usually* chosen to be aligned with the Greenwich meridian. The z_e axis projects from the earth's center of mass directly through the North pole. (This frame rotates at exactly the earth rate, ω_{ie} , about the z_e axis.)
3. Geographic (or navigation) frame: an orthogonal coordinate system; its origin is at the location of the INS (or the user), and its axes are aligned with the North, East, and Down directions [N,E,D]. This description implies that the INS platform is torqued to maintain the [N,E,D] orientation. (It must be noted that not all platforms are torqued in this manner. Some are allowed to "wander" about the z -axis, and are torqued in the level axes only. Platforms that are not torqued about the vertical axis are called *wander azimuth* systems [3].)
4. Body frame: an orthogonal frame; its origin is at vehicle (i.e., aircraft) center of mass. Its axes are the vehicle's roll, pitch, and yaw axes [R,P,Y]. Britting points out that the origin of the body frame rarely (if ever) coincides with the origin of the navigation frame [3].
5. Platform frame: an orthogonal frame that "can be thought of as three fiducial lines which are physically inscribed on the platform." [3]
6. Gyro frame: a (possibly) non-orthogonal frame; its axes are defined by the input axes of the three gyroscopes installed in the platform. The degree of non-orthogonality depends upon manufacturing and physical installation considerations [3, 16].
7. Accelerometer frame: a (possibly) non-orthogonal frame; its axes are defined by the three input axes of the accelerometers installed in the platform. Again, the degree of non-orthogonality depends on manufacturing and installation considerations [3, 16].

With Britting's definitions in hand, the Litton reference frame definitions are somewhat easier to decipher. The Litton reference frames as specified in [20] are interpreted with the help of the Britting definitions as follows:

1. Earth-Centered-Earth-Fixed (ECEF) frame: an orthogonal R^3 coordinate system; its origin is coincident with the earth's center of mass. The z_e , x_e plane lies in the earth's equatorial plane and the x_e axis is chosen to be aligned with the Greenwich meridian. The y_e axis projects from the earth's center of mass directly through the North pole. (This frame rotates at exactly the earth rate, ω_{ie} , about the y_e axis.)
NOTE: This frame is not the ECEF frame described by Britting. The transformation between the two frames is:

$$\begin{Bmatrix} x_e \\ y_e \\ z_e \end{Bmatrix}_{LITTON} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & -1 \\ 1 & 0 & 0 \end{bmatrix} \begin{Bmatrix} x_c \\ y_c \\ z_c \end{Bmatrix}_{BRITTING} \quad (2.1)$$

2. True frame: a level R^3 coordinate system located at the user's actual latitude and longitude. If $\alpha_t = 0^\circ$, then the x_t axis points to true East, the y_t axis points to true North, and the z_t axis points exactly Up, resulting in the Litton true frame $[E, N, U]_t$. (This is the error-free case of Britting's navigation reference frame.)
3. Computer frame: a level R^3 coordinate system located at the user's indicated latitude and longitude. If $\alpha_c = 0^\circ$, then the x_c axis points East, the y_c axis points North, and the z_c axis points Up, resulting in the Litton computer frame $[E, N, U]_c$. This frame is defined by coordinates which are calculated by the computer that is integral to the INS. Because of the presence of uncertainties in the computer frame calculations, discrepancies exist between quantities (i.e. latitude and longitude) calculated in this frame and the same quantities in the true frame.
4. Platform frame: this frame is important in the system error dynamic equations where the platform [mis]alignment with respect to the true frame must be taken into account. The result of *physically* installing an INS (situating the "black box" in an aircraft, for example) is often a slight misalignment of the platform frame with

respect to the true frame. The resulting error angles are defined by a skew-symmetric transformation ϕ :

$$\begin{Bmatrix} x \\ y \\ z \end{Bmatrix}^p = [I + \Phi] \begin{Bmatrix} x \\ y \\ z \end{Bmatrix}^t \quad (2.2)$$

where

$$\Phi = \begin{bmatrix} 0 & \phi_z & -\phi_y \\ -\phi_z & 0 & \phi_x \\ \phi_y & -\phi_x & 0 \end{bmatrix} \quad (2.3)$$

5. Body frame: Litton defines the body frame in identically the same way as does Britting. For the LN-93, the sensor frame is defined to be identically the same as the body frame.
6. Gyro frame: Litton also uses the Britting definition for the gyro frame. This frame is important in the system error dynamic equations where gyro misalignments with respect to the navigation frame (true and/or computer frame) must be taken into account. The physical construction of the INS "black box," including strapdown mechanizations using RLGs, is never perfect; the result is that the gyro frame is non-orthogonal to a certain extent [3, 18] although this is negligible for some applications [18].

Note that Litton assumes the body, sensor, and platform frames are coincident [20]. The same assumption is maintained for this study because INS platform location with respect to the body frame is specific to each aircraft type. When the platform and body frame origins are NOT coincident (as in most applications) the "lever arm" effect discussed in [31] must be taken into account. However, in the interest of generality and efficiency, the lever arm effect is not considered in this study. The gyro and accelerometer frame errors are considered and are appropriately reflected in the LN-93 error model.

2.4.1 Reference Frame Transformation Matrices. The LN-93 INS is a local-level, wander azimuth platform. Consequently, user position is specified in terms of latitude,

longitude, altitude, and wander angle. Generally, wander angle is the angle about the n -frame z axis which results because the INS platform is not torqued to maintain precise [ENU] orientation. In this mechanization, the platform is "torqued" only in the "level" axes in order to maintain the local level orientation. (The LN-93 is a strapdown mechanization which employs RLGs. Consequently, "torquing" takes place only in software.)

It is often necessary to express vectors such as position, attitude, velocity, or acceleration in terms of several different reference frames. As an example, the Litton LN-93 INS error-state model describes position error in terms of an *error-angle space* vector, $[\delta\theta_x, \delta\theta_y, \delta\theta_z, \delta h]^T$, where $\delta\theta_x$ is the error angle about the local level E axis, $\delta\theta_y$ is the error angle about the local level N axis, $\delta\theta_z$ is the error about the local level U axis, and δh is the altitude error. Even though Litton's definition is clear, if the error-angle vector is to have *physical* meaning, it must be transformed into a vector in *navigation error space*, $[\delta\phi, \delta\lambda, \delta\alpha, \delta h]^T$, where $\delta\phi$ is the error in latitude, $\delta\lambda$ is the longitude error, $\delta\alpha$ is the wander azimuth error, and δh is again the altitude error. A *transformation matrix*, C'' , permits compact transformation of the error-angle vector into an equivalent expression in navigation error space.

Britting developed general transformation matrices for many reference frame pairs. Those transformation matrices which are of primary interest in this research are shown below. The first four matrices are attributed to Britting [3] and the last matrix is developed by Litton [20]:

C_t^e	\triangleq	true-to-earth
C_n^e	\triangleq	navigation-to-earth
C_t^n	\triangleq	true-to-navigation
C_t^b	\triangleq	true-to-body
C_n''	\triangleq	error angle-to-navigation

Usage of these transformation matrices is illustrated in the following example in which a vector which is written in the j -frame is transformed to an equivalent expression in the

k -frame:

$$\begin{Bmatrix} X \\ Y \\ Z \end{Bmatrix}^k = \mathbf{C}_j^k \begin{Bmatrix} X \\ Y \\ Z \end{Bmatrix}^j \quad (2.4)$$

where the j and k subscripts and superscripts are replaced by the i , e , r , or t index as appropriate for the situation. Mathematical definitions for the transformation matrices defined above are shown below. For all cases in which they occur, λ is terrestrial longitude, L is geodetic latitude, and α_t is the true wander angle.

$$\mathbf{C}_t^r = \begin{bmatrix} \cos \lambda \cos \alpha_t - \sin L \sin \lambda \sin \alpha_t & -(\cos \lambda \sin \alpha_t + \sin L \sin \lambda \cos \alpha_t) & \sin \lambda \cos L \\ \cos L \sin \alpha_t & \cos L \cos \alpha_t & \sin L \\ -(\sin \lambda \cos \alpha_t + \cos \lambda \sin L \sin \alpha_t) & \sin \lambda \sin \alpha_t - \cos \lambda \sin L \cos \alpha_t & \cos \lambda \cos L \end{bmatrix} \quad (2.5)$$

$$\mathbf{C}_u^e = \begin{bmatrix} \cos \lambda & -\sin \lambda \sin L & \sin \lambda \cos L \\ 0 & \cos L & \sin L \\ -\sin \lambda & -\cos \lambda \sin L & \cos \lambda \cos L \end{bmatrix} \quad (2.6)$$

$$\mathbf{C}_t^i = \begin{bmatrix} \cos \alpha_t & -\sin \alpha_t & 0 \\ \sin \alpha_t & \cos \alpha_t & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.7)$$

$$\mathbf{C}_t^h = \begin{bmatrix} \cos \theta \sin \psi & \cos \theta \cos \psi & \sin \theta \\ \sin \phi \sin \theta \sin \psi + \cos \phi \cos \psi & \sin \phi \sin \theta \cos \psi - \cos \phi \sin \psi & -\sin \phi \cos \theta \\ \cos \phi \sin \theta \sin \psi - \sin \phi \cos \psi & \cos \phi \sin \theta \cos \psi + \sin \phi \cos \psi & -\cos \phi \cos \theta \end{bmatrix} \quad (2.8)$$

In the equation for \mathbf{C}_t^h above, ϕ is the aircraft roll angle, θ is the aircraft pitch angle, and ψ is the aircraft heading angle.

To conclude the discussion on reference frame transformations, the matrix \mathbf{C}_u^n is presented. \mathbf{C}_u^n converts a vector written in the Litton error-angle space, $[\delta\theta_x, \delta\theta_y, \delta\theta_z, \delta h]^T$ into one in navigation error space $[\delta\phi, \delta\lambda, \delta\alpha, \delta h]^T$. The transformation matrix is shown

below:

$$C_u^n = \begin{bmatrix} -\cos \alpha & \sin \alpha & 0 & 0 \\ \sin \alpha \sec \phi & \cos \alpha \sec \phi & 0 & 0 \\ -\sin \alpha \tan \phi & -\cos \alpha \tan \phi & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.9)$$

This matrix is used in the transformation

$$\begin{Bmatrix} \delta \phi \\ \delta \lambda \\ \delta \alpha \\ \delta h \end{Bmatrix} = \begin{bmatrix} C_u^n \end{bmatrix} \begin{Bmatrix} \delta \theta_u \\ \delta \theta_\lambda \\ \delta \theta_\alpha \\ \delta h \end{Bmatrix} \quad (2.10)$$

2.5 WGS-84 Geodetic System

In light of the numerous reference frames which are discussed in the previous section, it is apparent that the designer may choose any one of a number of coordinate systems in which to define position. The current most popular geodetic reference frame (ECEF) is defined by the World Geodetic Survey of 1984 (WGS-84). According to Kumar [15], the WGS-84 reference is the state-of-the-art in geodesy, reflecting the latest and best modeling techniques for characterizing the earth's surface.

The WGS-84 model is used exclusively as the ECEF reference for this thesis effort. Several of the key parameters from the standard are included in Table 2.1 [9, 31]. The X_c , Y_c , and Z_c coordinates are simply those which define the orthogonal ECEF axes of the WGS-84 model. The earth's angular rate is represented by ω_{ie} , and the equatorial and polar radius of the spheroid are given by A and B , respectively. The flattening parameter is defined as [3]:

$$f = \frac{A - B}{A} \quad (2.11)$$

where A and B are defined above, and eccentricity is [3]:

$$e = \sqrt{1 - \frac{B^2}{A^2}} \quad (2.12)$$

Table 2.1. WGS 84 Parameters

Parameter	Definition	Value
X_0, Y_0, Z_0	ECEF Coordinate Frame Axes	not applicable
ω_{ie}	Angular Rate of the Earth	$7.292115 \cdot 10^{-5} \text{ s}^{-1}$
A	Semimajor Axis (Equatorial Radius)	6378137 m
B	Semiminor Axis (Polar Radius)	6356752.3142 m
e	First Eccentricity	0.0818191908426
f	Flattening (Ellipticity)	0.00335281066474
g_0	Equatorial Acceleration of Gravity	$9.7803267714 \text{ m/s}^2$ ($32.087686258 \text{ ft/s}^2$)

2.6 Kalman Filter Theory

Most estimation problems which are not purely academic are fraught with uncertainty. The problem of navigation is a superb example. A navigation "solution" is obtained by solving some rather complex equations which depend on several quantities that are known to varying degrees of accuracy. As a result, the accuracy of the solution is limited. It is the role of the Kalman filter to account for the uncertainties associated with such a problem, and to provide an optimal solution (under certain conditions). Although it is far beyond the scope of this thesis to provide extensive details on the topic of Kalman filtering, it is essential to present a small amount of the basic theory along with the equations which describe the Kalman filter algorithm.

Maybeck describes a Kalman filter as an "optimal recursive data processing algorithm" which depends upon three basic assumptions [22]. If a *linear* (or linearized) system is driven by *white, Gaussian* noise, the Kalman filter solution is "optimal with respect to virtually any criterion [22]."

By optimally combining data (measurements), dynamic characteristics and statistical properties of the system and measuring devices, and initial conditions, the Kalman filter

produces estimates of the modeled quantities which are more accurate than estimates based on the individual measurements. The development which follows is a summary of important Kalman filtering concepts as presented by Maybeck [22, 23].

2.6.1 Linear Kalman Filtering. Before Kalman filtering is undertaken, the designer must develop a mathematical description for the system of interest. When it is possible to do so, a system is modeled as a set of linear differential equations of the form [22]:

$$\dot{\mathbf{x}}(t) = \mathbf{F}(t)\mathbf{x}(t) + \mathbf{B}(t)\mathbf{u}(t) + \mathbf{G}(t)\mathbf{w}(t) \quad (2.13)$$

where

\mathbf{x}	=	"state" vector
\mathbf{F}	=	homogeneous state dynamics matrix
\mathbf{B}	=	control input matrix
\mathbf{u}	=	deterministic control input vector
\mathbf{G}	=	driving noise input matrix
\mathbf{w}	=	white Gaussian driving noise vector

Because the deterministic control term $\mathbf{B}\mathbf{u}$ is zero in this research, it will be ignored hereafter. Also note that for this thesis, $\mathbf{G} = \mathbf{I}$. The mean of the white Gaussian driving noise vector is:

$$\mathbf{m}_w = E\{\mathbf{w}(t)\} = 0 \quad (2.14)$$

and the noise strength is $\mathbf{Q}(t)$:

$$E\{\mathbf{w}(t)\mathbf{w}^T(t + \tau)\} = \mathbf{Q}(t)\delta(\tau) \quad (2.15)$$

While Equation (2.13) is written in terms of 'whole' value state variables, the models used in this thesis are those of error states. This choice of state variables results in simpler dynamic equations [3], and Equation (2.13) may be rewritten as [22]:

$$\delta\dot{\mathbf{x}}(t) = \mathbf{F}(t)\delta\mathbf{x}(t) + \mathbf{B}(t)\mathbf{u}(t) + \mathbf{G}(t)\mathbf{w}(t) \quad (2.16)$$

where $\mathbf{x}(t)$ has been replaced by the error state vector $\delta\mathbf{x}(t)$, and all other quantities retain their previous definitions. The topic of error states is explored more fully in the section on extended Kalman filters.

As previously stated, the Kalman filter incorporates measurement information from external measuring devices. Irrespective of the type of measuring device, the equation which is used to describe linear measurements is of the form:

$$\mathbf{z}(t_i) = \mathbf{H}(t_i)\mathbf{x}(t_i) + \mathbf{v}(t_i) \quad (2.17)$$

or, in the case of error-state models:

$$\delta\mathbf{z}(t_i) = \mathbf{H}(t_i)\delta\mathbf{x}(t_i) + \mathbf{v}(t_i) \quad (2.18)$$

where, in both cases above, \mathbf{H} is the *observation* matrix and \mathbf{v} is a discrete-time zero-mean white Gaussian measurement noise vector whose covariance is [22]:

$$E\{\mathbf{v}(t_i)\mathbf{v}^T(t_j)\} = \begin{cases} \mathbf{R} & \text{for } t_i = t_j \\ \mathbf{0} & \text{for } t_i \neq t_j \end{cases} \quad (2.19)$$

The Kalman filter "propagates" the error state and its covariance from the instant in time immediately following the most recent measurement update, t_i^+ , to the instant in time immediately preceding the next measurement update, t_{i+1}^- by numerical integration of the following equations [22]:

$$\dot{\hat{\mathbf{x}}}(t/t_i) = \mathbf{F}(t)\hat{\mathbf{x}}(t/t_i) \quad (2.20)$$

$$\dot{\mathbf{P}}(t/t_i) = \mathbf{F}(t)\mathbf{P}(t/t_i) + \mathbf{P}(t/t_i)\mathbf{F}^T(t/t_i) + \mathbf{G}(t)\mathbf{Q}(t)\mathbf{G}^T(t) \quad (2.21)$$

with initial conditions:

$$\hat{\mathbf{x}}(t/t_i) = \hat{\mathbf{x}}(t_i^+) \quad (2.22)$$

$$\mathbf{P}(t/t_i) = \mathbf{P}(t_i^+) \quad (2.23)$$

as provided by the measurement update cycle at time t_i .

After propagation, $\hat{x}(t)$ and $P(t)$ are "updated" (meaning that state estimates are revised based on new information). The pivotal element in the update equations shown below is the time varying Kalman filter gain $K(t)$. The $K(t)$ matrix assigns "weights" to the "new information" (new information consists of the difference between the actual measurement and the filter's estimate of the measurement) based on known measurement noise statistics and filter-computed covariances from the previous time step. This process is designed to improve the estimate of each element of the state vector. The update equations are [22]:

$$K(t_i) = P(t_i^-)H^T(t_i)[H(t_i)P(t_i^-)H^T(t_i) + R(t)]^{-1} \quad (2.24)$$

$$\hat{x}(t_i^+) = \hat{x}(t_i^-) + K(t_i)[z_i - H(t_i)\hat{x}(t_i^-)] \quad (2.25)$$

$$P(t_i^+) = P(t_i^-) - K(t_i)H(t_i)P(t_i^-) \quad (2.26)$$

Although the algorithm shown above is generally applicable to any problem which lends itself to a Kalman filtering solution, it is not necessarily the algorithm which is used in practice. It is often advantageous to use a form of the algorithm known as the *U-D* form. In the *U-D* algorithm, the filter covariance matrix is not propagated as a square array. The matrices below representing the pre- and post-measurement filter covariances, respectively, are explicitly computed instead [22]:

$$P(t_i^-) = U(t_i^-)D(t_i^-)U(t_i^-) \quad (2.27)$$

$$P(t_i^+) = U(t_i^+)D(t_i^+)U(t_i^+) \quad (2.28)$$

where the U matrices are upper triangular and unitary (and thus contain ones along the main diagonal), and the D matrices are simply diagonal [22]. This form offers several advantages including numerical stability, improved precision, and guaranteed nonnegativity of the computed covariances [22]. It is the *U-D* form of the Kalman filter algorithm which is implemented in the MSOFE software [5].

2.6.2 Linearized and Extended Kalman Filtering. Unfortunately, not all problems are adequately described as linear systems driven by white Gaussian noise. In many cases, the most appropriate model is nonlinear. The navigation problem at hand falls squarely into the nonlinear category. Fortunately, a method exists whereby a nonlinear system may be treated in much the same manner as a linear one for a particular class of problems. Suppose that the nonlinear system may be described by [23]:

$$\dot{\mathbf{x}}(t) = \mathbf{f}[\mathbf{x}(t), \mathbf{u}(t), t] + \mathbf{G}(t)\mathbf{w}(t) \quad (2.29)$$

In this case, the state dynamics matrix, $\mathbf{f}[\cdot, \cdot, \cdot]$, is a nonlinear function of the state vector $\mathbf{x}(\cdot)$, time t , and the control input (assumed to be zero in this research). The white Gaussian noise is defined exactly as in Equations (2.14) and (2.15). In addition, the measurement equation may also be a nonlinear function of time [23]:

$$\mathbf{z}(t_i) = \mathbf{h}[\mathbf{x}(t_i), t_i] + \mathbf{v}(t_i) \quad (2.30)$$

The noise vector \mathbf{v} is again zero-mean and its covariance is described by Equation (2.19).

Recalling that a system must be linear in order to satisfy the assumptions that yield the optimality of a Kalman filter, the nonlinear Equations (2.29) and (2.30) must be linearized. The following approach is summarized from Maybeck [23]:

1. Assume that a nominal state trajectory, $\mathbf{x}_n(t)$, may be generated which satisfies $\mathbf{x}_n(t_0) = \mathbf{x}_{n0}$ and

$$\dot{\mathbf{x}}_n(t) = \mathbf{f}[\mathbf{x}_n(t), \mathbf{u}(t), t] \quad (2.31)$$

where $\mathbf{f}[\cdot, \cdot, \cdot]$ is specified in Equation (2.29), and $\mathbf{u}(t) = \mathbf{0}$.

2. The "nominal" measurements which accompany the nominal trajectory are:

$$\mathbf{z}_n(t_i) = \mathbf{h}[\mathbf{x}_n(t_i), t_i] \quad (2.32)$$

3. The "perturbation" of the state is obtained by subtracting the nominal trajectory from the original nonlinear equation:

$$[\dot{\mathbf{x}}(t) - \dot{\mathbf{x}}_n(t)] = \mathbf{f}[\mathbf{x}(t), \mathbf{u}(t), t] - \mathbf{f}[\mathbf{x}_n(t), \mathbf{u}(t), t] + \mathbf{G}(t)\mathbf{w}(t) \quad (2.33)$$

4. The equation above may be approximated to first order by a Taylor series expansion:

$$\delta \dot{\mathbf{x}}_n(t) = \mathbf{F}[t; \mathbf{x}_n(t)] + \mathbf{G}(t)\mathbf{w}(t) \quad (2.34)$$

where $\delta \mathbf{x}(t)$ represents a first-order approximation of the process $[\mathbf{x}(\cdot) - \mathbf{x}_n(\cdot)]$, and $\mathbf{F}[t; \mathbf{x}_n(t)]$ is a matrix of partial derivatives of \mathbf{f} with respect to its first argument, evaluated along the nominal trajectory [23]:

$$\mathbf{F}[t; \mathbf{x}_n(t)] = \left. \frac{\partial \mathbf{f}[\mathbf{x}(t), t]}{\partial \mathbf{x}} \right|_{\mathbf{x}=\mathbf{x}_n(t)} \quad (2.35)$$

5. The perturbation measurement equation is derived in like fashion and is expressed as [23]:

$$\delta \mathbf{z}_n(t) = \mathbf{H}[t; \mathbf{x}_n(t)] + \mathbf{v}(t_i) \quad (2.36)$$

where

$$\mathbf{H}[t; \mathbf{x}_n(t)] = \left. \frac{\partial \mathbf{h}[\mathbf{x}(t), t]}{\partial \mathbf{x}} \right|_{\mathbf{x}=\mathbf{x}_n(t)} \quad (2.37)$$

With the "error-state" model in hand, it is possible to return to the linear filtering theory. An estimate of the whole-valued quantities of interest is obtained from [23]:

$$\hat{\mathbf{x}}(t) = \mathbf{x}_n(t) + \delta \hat{\mathbf{x}}(t) \quad (2.38)$$

The expression above for the linearized Kalman filter is useful provided that the linearization assumption is not violated. However, if the nominal and "true" trajectories differ by too large an amount, unacceptable errors may result [23]. It is for this reason that extended Kalman filtering is useful in many cases where perturbation techniques alone do not suffice. Extended Kalman filtering allows for *relinearizing* about newly declared

nominals at each sample time, to enhance the adequacy of the linearization process, and thus of the resulting filter performance as well [25].

The extended Kalman filter equations are summarized below. The reader is referred to Maybeck [23] for the details of derivation. The measurement equation for an extended Kalman filter is:

$$z(t_i) = h[x(t_i), t_i] + v(t_i) \quad (2.39)$$

where $v(\cdot)$ is once again zero-mean with covariance given by Equation (2.19). Measurements are incorporated into the extended Kalman filter via the following set of equations [23]:

$$K(t_i) = P(t_i^-) H^T[(t_i); \hat{x}(t_i^-)] \{ H[(t_i); \hat{x}(t_i^-)] P(t_i^-) H^T[(t_i); \hat{x}(t_i^-)] + R(t_i) \}^{-1} \quad (2.40)$$

$$\hat{x}(t_i^+) = \hat{x}(t_i^-) + K(t_i) \{ z_i - h[\hat{x}(t_i^-); (t_i)] \} \quad (2.41)$$

$$P(t_i^+) = P(t_i^-) - K(t_i) H[(t_i); \hat{x}(t_i^-)] P(t_i^-) \quad (2.42)$$

where

$$H[t_i; \hat{x}(t_i^-)] = \left. \frac{\partial h[x, t_i]}{\partial x} \right|_{x=\hat{x}(t_i^-)} \quad (2.43)$$

The state estimate and covariance are propagated from t_i to t_{i+1} by integrating the following equations [23]:

$$\dot{\hat{x}}(t/t_i) = f[\hat{x}(t/t_i), u(t), t] \quad (2.44)$$

$$\dot{P}(t/t_i) = F[t; \hat{x}(t/t_i)] P(t/t_i) + P(t/t_i) F^T[t; \hat{x}(t/t_i)] + G(t) Q(t) G^T(t) \quad (2.45)$$

where

$$F[t; \hat{x}(t/t_i)] = \left. \frac{\partial f[x(t), u(t), t]}{\partial x} \right|_{x=\hat{x}(t/t_i)} \quad (2.46)$$

and the initial conditions are:

$$\hat{x}(t/t_i) = \hat{x}(t_i^+) \quad (2.47)$$

$$P(t/t_i) = P(t_i^+) \quad (2.48)$$

The equations shown above for the extended Kalman filter are programmed into the

MSOPE shell for the problem defined by this thesis. It is the fact that the extended Kalman filter is relinearized about each successive estimate of the state $\hat{\mathbf{x}}(t)$ which "enhances the validity of the assumption that deviations from the reference (nominal) trajectory are small enough to allow linear perturbation techniques to be employed" [23].

2.7 Summary

This chapter introduces several topics which are related to the NRS integration problem. Major systems such as the INS, the RRS Transponder system, and the GPS are described briefly and their roles as subsystems in the NRS are outlined. Reference frames are discussed in moderate detail and several important frames are defined. Coordinate transformations are also supplied. The WGS-84 Geodetic System is presented and key parameters are tabularized. Finally, a very brief outline of some of the Kalman filtering theory concepts crucial to this thesis are presented. Equations for the linearized and extended Kalman filters are shown and the validity of using the extended Kalman filter algorithm for the nonlinear problem in this thesis is explained.

III. INS and Baro-Altimeter Models

3.1 Introduction

The INS chosen for use in the proposed Navigation Reference System (NRS) is the Litton LN-93. The LN-93 is a state-of-the-art strapdown INS which uses RLGs in its mechanization rather than the older gimbaled gyros. The LN-93 characteristics promise a considerable advantage over mechanizations used in previous navigation standards such as CIRIS (i.e., the LN-15 used in CIRIS I and the LN-39 used in CIRIS II). The LN-93 error model is presented below. Additionally, the need for a revision to the baro-altimeter model is presented and the revised model is developed.

3.2 Litton LN-93 INS

Litton specifications stipulate that the LN-93 horizontal velocity errors must be smaller than 2.5 feet per second (RMS) after two hours of navigation following a gyrocompass alignment, and that horizontal position error must be less than 0.8 nautical miles per hour during the same period [20]. Actual statistics provided in the Litton documentation show that for a period of ten hours of (static) navigation, the LN-93 is accurate to within 2 nautical miles. For a 2 hour "flight" simulating a fixed flight profile (which is discussed in Section 3.4), the LN-93 maintains horizontal position accuracy to better than 1 nautical mile.

Figure 3.1 depicts the basic baro-aided INS subsystem. The INS subsystem, aided by a baro-altimeter, forms the foundation upon which the full NRS model is built. NRS will eventually incorporate RRS and GPS models, as presented in Chapters IV and V, respectively.

3.2.1 Error Types. Litton's comprehensive dynamic error model for the LN-93 is composed of 93 states. The error sources modeled fall into one of several error source types. First, the random constant represents a variable which may take on any value within prescribed limits (defined by a random value chosen from a Gaussian distribution of appropriate scale) and thereafter retains its initial value [22]. Variables of this type are

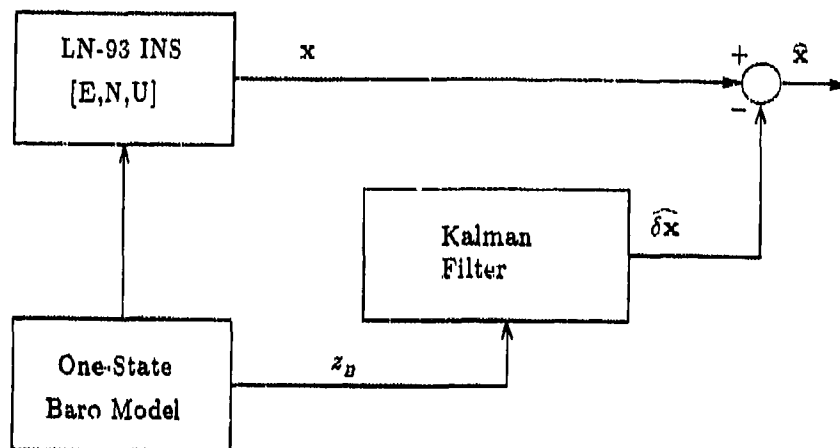


Figure 3.1. Basic Baro-Aided LN-93 INS

often referred to as biases and are modeled mathematically by [22]:

$$\dot{\hat{x}}(t) = 0 \quad , \quad \text{and} \quad P(t) = P_0 \quad (3.1)$$

Approximately 64 bias error state variables are included in the Litton model. These are lumped into subcategories, including gyro bias states, accelerometer bias states, trend states, and gyro compliance states.

An error variable type *not* represented in the LN-93 model is the random walk or Brownian motion variable. Although the Litton model does not include such variables, Maybeck recommends that variables of this type be used in the Kalman filter implementation of random bias states when there exists a tendency for the covariances of those states to drop to zero [22]. This practice (adding small magnitude "pseudo-noises" to states whose truth model includes no noise source) prevents covariances of those states from dropping to zero, thereby improving the estimation potential for those states. Variables of this type are characterized by a tendency to wander about in an unpredictable fashion, and are described as "the output of an integrator driven by white Gaussian noise" [22].

They are represented mathematically by [22]:

$$\dot{x}(t) = w(t) \quad , \quad \text{and} \quad P(t) = P_0 \quad (3.2)$$

where w represents zero-mean white Gaussian noise of strength q . Variables of this type are not used in this error model. However, they are generally useful when Kalman filters containing bias states are implemented [22].

Another error source type included in the LN-93 model is the first-order Markov process. This variable is also characterized by a tendency to wander about, but with a certain degree of correlation from one time to another. This process is described as the output of a first order lag which is driven by zero-mean white Gaussian noise [22]. The first-order Markov variable is modeled by an integrator with a negative feedback loop. The feedback gain is the reciprocal of the time constant for the variable of interest [22]. Its mathematical representation is [22]:

$$\dot{x}(t) = -\frac{1}{\tau} x(t) + w(t) \quad , \quad \text{and} \quad P(t) = P_0 \quad (3.3)$$

where w represents zero-mean white Gaussian noise of strength q , and τ is the time constant associated with the state of interest. Error variables which are included in this category are gyro and accelerometer drifts and thermal transients associated with these inertial instruments. (Additionally, the baro-altimeter state included by Litton falls into this category. This is discussed in more detail later.)

The last category of errors in the Litton model are those which are termed "general" [20]. These error states are complex combinations of several states. Some states which fall into this category depend on as many as 40 to 50 other states. Navigation position, velocity, and platform tilt error states are examples of such complex interdependency. The "lower" 13 states (termed the general error states) are coupled to one another and to other states. The Litton linearized error model is presented in the form of a general equation of the type presented in Chapter II. The generalized INS error state equation is of the form [3, 22]:

$$\delta \dot{\mathbf{x}}(t) = \mathbf{F}(t) \delta \mathbf{x}(t) + \mathbf{G} \mathbf{w}(t) \quad (3.4)$$

where

$$F(t) = \frac{\partial F[x, u(t), t]}{\partial x} \Big|_{x=x_n} \quad (3.5)$$

and $\delta x(t)$ is the 93×1 time varying state error vector, $F(t)$ is a 93×93 time varying system dynamics matrix, $u(t)$ is a zero vector, $w(t)$ is a 93×1 white noise vector, and $G = I$.

3.2.2 Error State Vector. Litton partitions the error state vector into six subvectors as shown below [20]. Note that each of the subvectors is expressed as one of the error types discussed above.

$$\delta x = [\delta x_1^T \quad \delta x_2^T \quad \delta x_3^T \quad \delta x_4^T \quad \delta x_5^T \quad \delta x_6^T]^T \quad (3.6)$$

where δx is a 93×1 column vector and:

δx_1 represents the "general" error vector containing position, velocity, attitude, and vertical channel errors.

δx_2 consists of gyro, accelerometer, and baro-altimeter correlated errors, and "trend" states. These states are modeled as first order Markov processes in both the truth (system) model and in the Kalman filter.

δx_3 represents gyro bias errors. These states are modeled as random constants in the truth model and are modeled as random walks (with small magnitude pseudo-noises) in the Kalman filter.

δx_4 is the accelerometer bias error states. These states are modeled in exactly the same manner as the gyro bias states.

δx_5 depicts accelerometer and gyro initial thermal transients. The thermal transient states are first order Markov processes in the system and Kalman filter.

δx_6 models the gyro compliance errors. These error states are modeled as biases in the system model and as random walks in the Kalman filter.

The system state space differential equation is given as:

$$\begin{Bmatrix} \delta \dot{x}_1 \\ \delta \dot{x}_2 \\ \delta \dot{x}_3 \\ \delta \dot{x}_4 \\ \delta \dot{x}_5 \\ \delta \dot{x}_6 \end{Bmatrix} = \begin{bmatrix} F_{11} & F_{12} & F_{13} & F_{14} & F_{15} & F_{16} \\ 0 & F_{22} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & F_{55} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{Bmatrix} \delta x_1 \\ \delta x_2 \\ \delta x_3 \\ \delta x_4 \\ \delta x_5 \\ \delta x_6 \end{Bmatrix} + \begin{Bmatrix} w_1 \\ w_2 \\ 0 \\ 0 \\ 0 \\ 0 \end{Bmatrix} \quad (3.7)$$

All states in the 93 state INS error vector are defined in Appendix A, Tables A.1 through A.4. All non-zero elements of the dynamics matrix, F , and the process noise matrix, Q , are obtained from [20] and are presented in Appendix B.

3.3 Baro-Altimeter Model Revision

Although the Litton error model is extensive and reasonably thorough in its depiction of inertial instrument error sources, it is somewhat inadequate in its depiction of the error sources related to the baro-altimeter. Other authors discuss several error sources normally linked with baro-altitude [16, 28, 34] which appear to have been omitted in the Litton LN-93 model. The LN-93 error model contains only a single state for the barometric altimeter. For this reason, a revised baro-altitude error model is developed and embedded in the INS error model. The new baro-altitude model includes states for correlated noise effects, bias error, and scale-factor error. The revised baro model coefficient for the correlated noise state is extracted from the Litton documentation [20]. In the case of the other (new) baro states, the coefficients are extracted from [34]. These error sources are combined in the manner discussed below to formulate a more complete error model upon which to base subsequent work.

3.3.1 Litton Vertical Channel Aiding. As noted above, the Litton LN-93 error model contains only one state with which to model error sources intrinsic to the baro-altimeter. The state is a first order Markov process, and is modeled mathematically as

[20]:

$$\dot{x}(t) = -\beta_{hh_c} x(t) + w(t) \quad (3.8)$$

where w represents zero-mean white Gaussian noise of strength $2(\beta_{hh_c} \times \sigma_{\delta h_c}^2)$, β_{hh_c} is the inverse correlation time for the baro-altimeter, and $\sigma_{\delta h_c}^2$ is the variance of the baro-altimeter correlated noise as specified by Litton [20]. Litton specifies a correlation time of 600 seconds and $\sigma_{\delta h_c} = 100.0 ft$ for the two hour fighter run. The initial covariance of the baro-altimeter state is assumed to be $P_0 = 10^{-1} ft^2$. The state is represented in Figure 3.2.

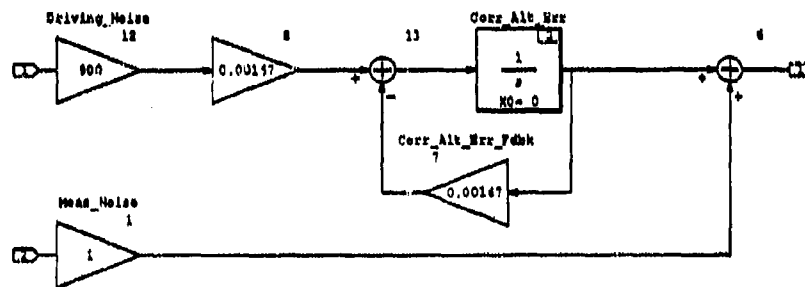


Figure 3.2. LN-93 Baro-Altitude Error Model

The single state baro model aids the vertical channel states (i.e. INS altitude and vertical velocity) in order to prevent the vertical channel instabilities discussed in Chapter II. The baro error state is coupled directly to four states in the dynamics equations, and is indirectly coupled to more than a dozen others. Figure 3.3 depicts the vertical channel error model included in the Litton reference [20]. The baro error δh_H enters the INS vertical channel error model on the right side near the middle of the diagram. It is notable that δh_H is approximated by δh_c , the correlated baro altitude error, in the Litton dynamics model. Thus, even though the Litton document discusses several error sources, (and in fact defines total baro-altimeter error to be a function of bias error, scale factor error, and correlated noise error) these error sources are not modeled (without explanation) and a single state δh_c , is used to represent the baro-altimeter model.

Because the altimeter error magnitude is significant to many of the INS states, the

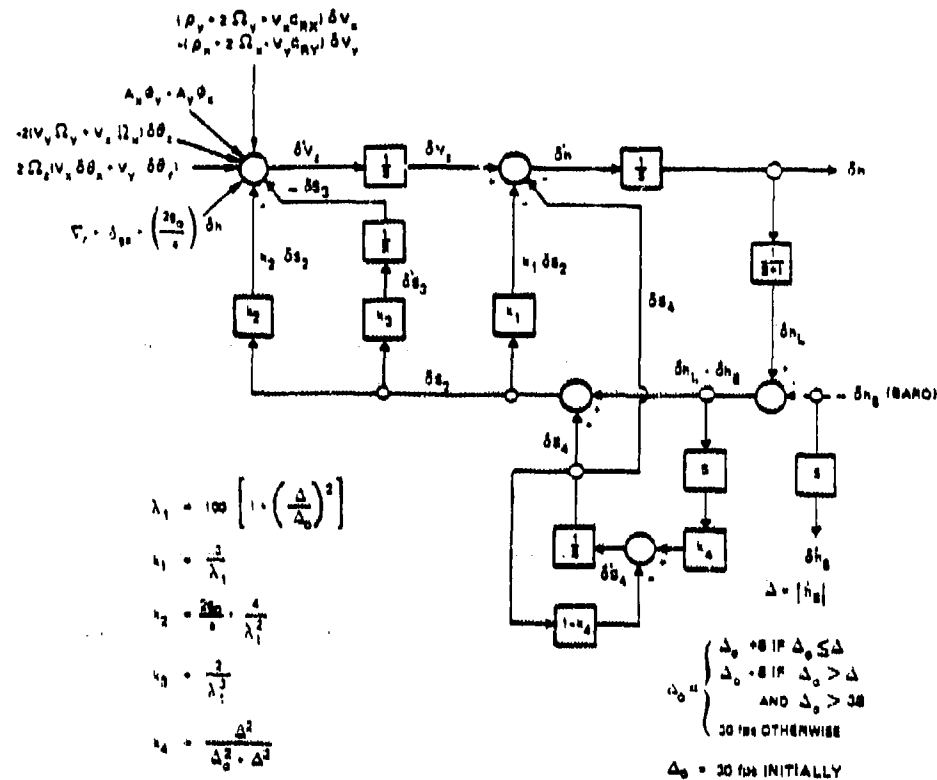


Figure 3.3. LN-93 Vertical Channel Error Model [20]

fidelity of the baro error model is of considerable concern. It is precisely to improve the its fidelity that the baro error model is revised as described in the next section.

3.3.2 Revised Baro-Altitude Model. The single baro state included in the Litton error model is adequate to represent one of the major sources of baro altitude error. Its relevant equations and parameters are exactly as shown in Equation (3.8). However, other baro-altimeter error sources are present and should be modeled in a high-fidelity truth model.

Another significant source of error in the baro-altimeter which must be taken into account is the error due to instrument bias. This error source is well modeled as a random constant whose initial one σ value is chosen to be 15 feet in this study. The choice is based

on a baro-altimeter model contained in [34].

Another significant error source included in the revised model is the scale factor error inherent in baro altimeters. This error is a multiplicative combination of aircraft altitude and the random constant scale factor state as shown below. The initial one σ value for the scale factor state is 0.01 [34]. Although the error contribution for this state is negligible at low altitudes, it can be significant source of error at higher altitudes.

The last error state in the revised baro-altimeter model is the first order lag which sums and filters the previously described errors. The output of this state represents the total, lagged baro-altimeter error δh_B . It is shown along with the other revised baro model states in Figure 3.4.

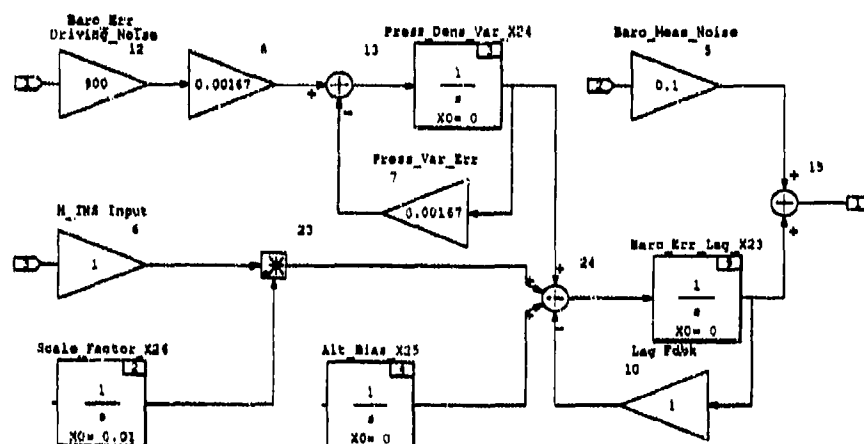


Figure 3.4. Revised Baro-Altitude Error Model

The revised baro-altimeter model shown in Figure 3.4 is incorporated into the NRS model. It occupies states 23, 24, 25, and 26 in the NRS state vector as shown in Appendix

A. State 23 is the total baro-altimeter error, δh_H . State 24 is the correlated baro altitude error, state 25 is the bias error state, and state 26 is the scale factor error state. The performance and validation of the revised baro model are discussed in Chapter VI.

3.4 Simulated Fighter Flight Profile

Litton has analyzed the error model for the LN-93 in several modes [20]. One important example is performance analysis of the LN-93 when it is "stressed" in a manner typical of a fighter mission flight profile. In other words, when the LN-93 is subjected to rapid acceleration, velocity, and attitude changes typical of a fighter mission flight profile, what are the performance characteristics of the INS? In an attempt to answer this question, the flight profile shown in Figure 3.4 is simulated in software.

Trajectory data for the flight profile shown in Figure 3.4 are calculated using PROF-GEN [1]. The trajectory data include variables such as latitude, longitude, altitude, accelerations, velocities, and attitude. These data are used in the MSOFE [5] simulation as "nominal" quantities about which the truth (system) model is relinearized after each integration step [22]. The extended Kalman filter is relinearized about its best estimate of the trajectory quantities.

3.5 Summary

This chapter introduces the main error model types which are generally important to Kalman filtering and specifically important to the Litton LN-93 error model. The LN-93 error model, which consists of 93 states as provided by Litton, is introduced. Because it has only one baro-altimeter error state, the LN-93 error model is revised to improve its fidelity; as a result, a four-state baro-altimeter model is used in the overall INS model. The flight profile which is used for subsequent Kalman filter evaluation is presented and briefly discussed.

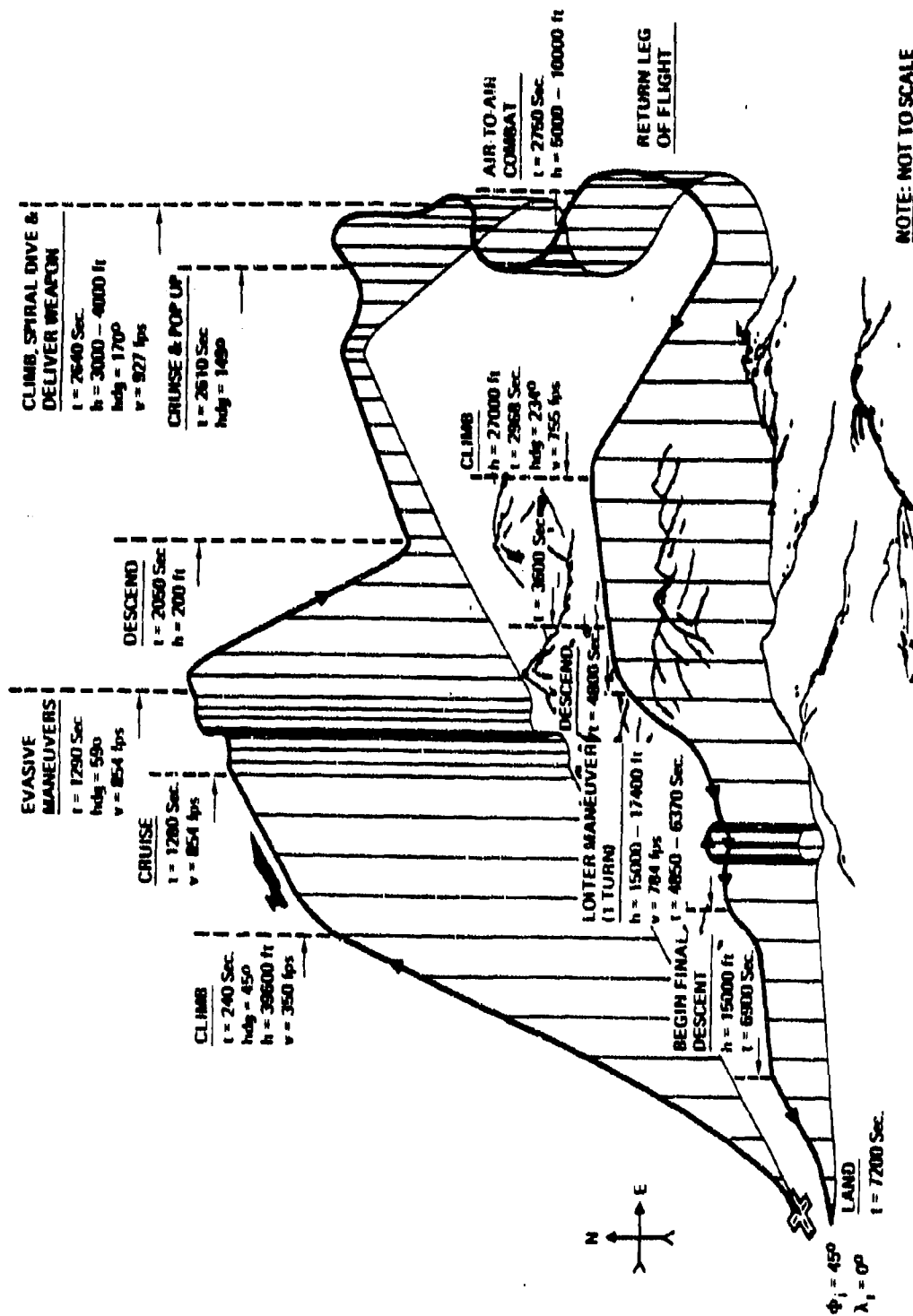


Figure 3.5. Fighter Mission Flight Profile [20]

IV. Range/Range-Rate Transponder System

4.1 Introduction

The Range/Range-Rate System (RRS) is a navigation aiding system which comprises a significant part of CIRIS and ARS. Consequently, the RRS is used in the Navigation Reference System (NRS) as well. Navigation information is obtained by "interrogating" ground transponders and subsequently processing the electromagnetic (EM) signals which the transponders emit. The information obtained allows high quality range and range-rate measurements to be calculated by the RRS interrogating hardware [31]. Using these range and range-rate measurements, refinements to the NRS position and velocity estimates are then possible.

The dynamic error model for the RRS transponder system is contained in [31, 33]. Portions of that work are summarized in this chapter. Note that the lever-arm effect discussed in [31] is not included in this work. Although the lever-arm effect is very important to consider in actual hardware applications, the assumption of INS and RRS interrogator antenna collocation does not reduce the validity of the conclusions which may be drawn from this work. The assumption of collocation is made in order to maintain generality but has the advantage of reducing computational loading. For applications in which the collocation assumption is not valid [such as the case in which actual data are used], a transformation must be applied to translate and rotate the RRS measurement into the navigation frame. One such transformation matrix is developed by Snodgrass [31]. However, the transformation matrix developed by Snodgrass [31] assumes a [N,W,U] navigation coordinate system (as used in the LN-39 INS) rather than the [E,N,U] system which is assumed in the Litton error model for the LN-93.

Figure (4.1) depicts the addition of the RRS transponder subsystem to the baro-aided INS diagram presented in Chapter III. This is another step in the path toward the full NRS model (which eventually incorporates GPS as well).

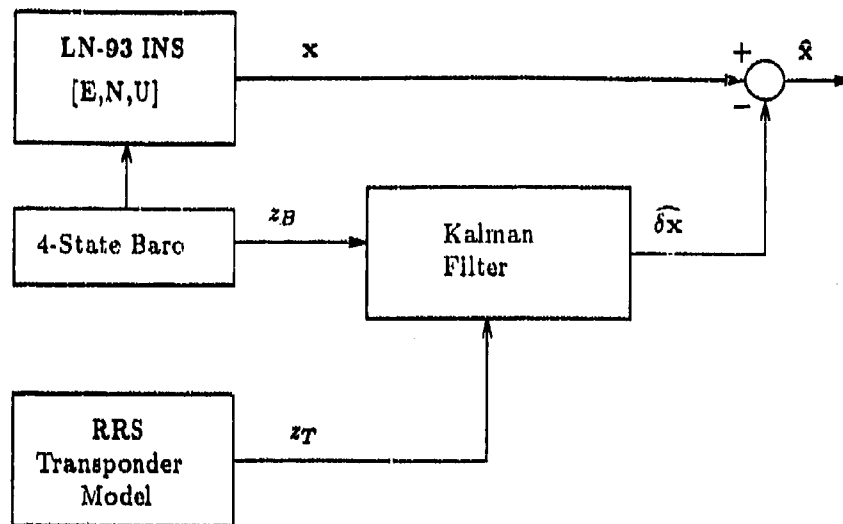


Figure 4.1. Baro-Aided INS with RRS Transponder Aiding

4.2 RRS Range Measurements

In CIRIS (or the proposed NRS), RRS range measurements aid in estimating position errors of the reference INS. The RRS range measurement is derived from the time delay detected between the time at which the reference hardware (CIRIS or NRS) sends an interrogation signal and the time at which a reply from the transponder is received. This temporal difference is multiplied by the speed of light [and divided by two, to account for the "round trip" of the electro-magnetic (EM) signal] to obtain an uncorrected range measurement. Correction factors are then applied in order to compensate for delays introduced by the propagation of EM signals through the atmosphere and to correct for errors introduced by equipment calibration biases [31].

Figure (4.2) shows an earth-user-transponder system in two dimensions. A single transponder is shown for clarity in this discussion. However, six to ten transponders are typically used during reference navigation system Kalman filter updating.

The true positions of the transponder and the user are shown along with the true range R , and the (uncorrected) range measurement R_{unc} (which is the range from the

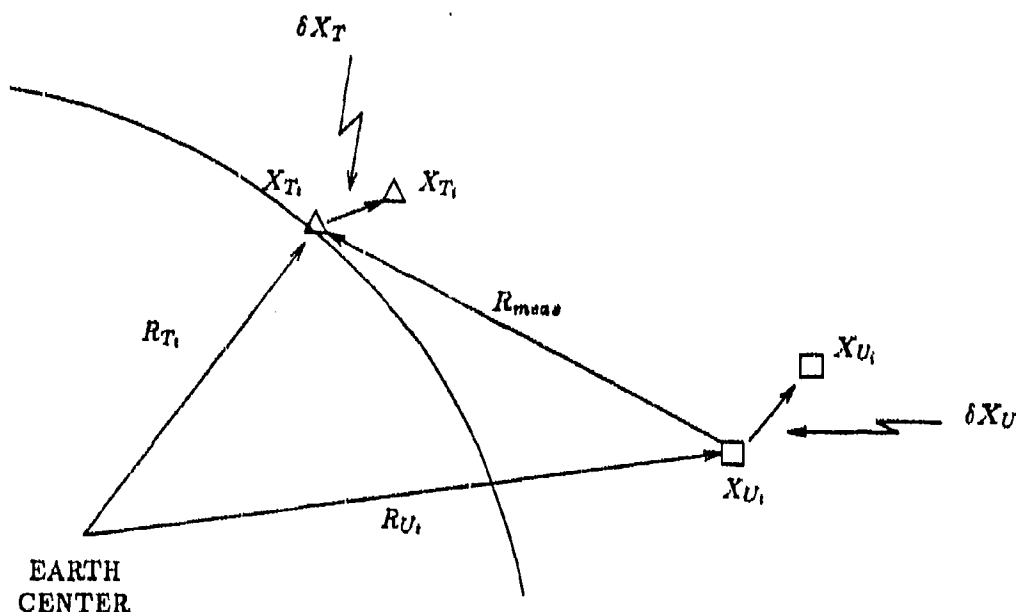


Figure 4.2. Two Dimensional Earth-User-Transponder System

ground transponder to the user, or more specifically, to the user's INS platform). It should be noted that both the true range and the uncorrected range are collinear with the true user and transponder positions. The vectors δX_T and δX_U are the transponder and user position error vectors, respectively. The moment-arm effect between the INS and RRS receiving antenna is not represented in the figure; it is assumed for this discussion that the RRS interrogator antenna is collocated with the INS. The (uncorrected) range measurement as obtained from RRS is:

$$R_{RRS} = R_t + \delta R_{atm} + \delta R_{br} + v \quad (4.1)$$

where

R_{RRS}	=	RRS range measurement, from transponder to user
R_t	=	True range, from transponder to user
δR_{atm}	=	Range error due to atmospheric delay
δR_{br}	=	Range error due to equipment calibration
v	=	zero-mean white Gaussian measurement noise

Equation (4.1) is a model for the range as determined by the RRS truth model. It includes the true range [which is never precisely known] along with terms which reflect sources of error and uncertainty. RRS error sources and models are discussed later.

4.3 Range Calculation From INS Data

In order to formulate a difference measurement as discussed in Chapter II, two sources of range information must be obtained. The first is the RRS range measurement which is modeled by Equation (4.1). Another range indication is computed from the INS indicated position and RRS (surveyed) positions. (Both the indicated INS position and transponder surveyed position contain uncertainties which must be considered.) In this approach, the user (INS) indicated position is represented by an R^3 vector expressed in the Litton ECEF as:

$$\mathbf{X}_U = \begin{Bmatrix} x_U \\ y_U \\ z_U \end{Bmatrix} \quad (4.2)$$

while the true RRS transponder position is represented in the Litton ECEF by:

$$\mathbf{X}_T = \begin{Bmatrix} x_T \\ y_T \\ z_T \end{Bmatrix} \quad (4.3)$$

Then the calculated range from the user (INS) to the transponder is given by:

$$R_{INS} = |\mathbf{X}_U - \mathbf{X}_T| = \left| \begin{Bmatrix} x_U \\ y_U \\ z_U \end{Bmatrix} - \begin{Bmatrix} x_T \\ y_T \\ z_T \end{Bmatrix} \right| \quad (4.4)$$

Equation (4.4) may be rewritten as:

$$R_{INS} = \sqrt{(x_U - x_T)^2 + (y_U - y_T)^2 + (z_U - z_T)^2} \quad (4.5)$$

If the equation above is "perturbed" to reflect uncertainties in user and transponder positions, then a first-order Taylor series may be written to approximate the range [3, 22]. The truncated (first-order) series is of the form [33]:

$$R_{INS} = R_t + \left. \frac{\partial R_{INS}(\mathbf{x}_T, \mathbf{x}_U)}{\partial \mathbf{x}_T} \right|_{(\mathbf{x}_T, \mathbf{x}_U)_{nom}} \cdot \delta \mathbf{x}_T + \left. \frac{\partial R_{INS}(\mathbf{x}_T, \mathbf{x}_U)}{\partial \mathbf{x}_U} \right|_{(\mathbf{x}_T, \mathbf{x}_U)_{nom}} \cdot \delta \mathbf{x}_U \quad (4.6)$$

After substituting Equation (4.5) into Equation (4.6) and evaluating the partial derivatives, the INS-derived range approximation becomes [31, 33]:

$$R_{INS} = R_t - \left[\frac{x_T - x_U}{|R_{INS}|} \right] \cdot \delta x_U - \left[\frac{y_U - y_U}{|R_{INS}|} \right] \cdot \delta y_U - \left[\frac{z_T - z_U}{|R_{INS}|} \right] \cdot \delta z_U + \left[\frac{x_T - x_U}{|R_{INS}|} \right] \cdot \delta x_T + \left[\frac{y_T - y_U}{|R_{INS}|} \right] \cdot \delta y_T + \left[\frac{z_T - z_U}{|R_{INS}|} \right] \cdot \delta z_T \quad (4.7)$$

At this point, the difference measurement may be formed as:

$$\begin{aligned} \delta z &= R_{INS} - R_{RRS} \\ &= - \left[\frac{x_T - x_U}{|R_{INS}|} \right] \cdot \delta x_U - \left[\frac{y_U - y_U}{|R_{INS}|} \right] \cdot \delta y_U - \left[\frac{z_T - z_U}{|R_{INS}|} \right] \cdot \delta z_U \\ &\quad + \left[\frac{x_T - x_U}{|R_{INS}|} \right] \cdot \delta x_T + \left[\frac{y_T - y_U}{|R_{INS}|} \right] \cdot \delta y_T + \left[\frac{z_T - z_U}{|R_{INS}|} \right] \cdot \delta z_T \\ &\quad - [1] \delta R_{atm} - [1] \delta R_{br} + v \end{aligned} \quad (4.8)$$

Note that the true whole-valued range (R_t) formerly present in both individual range measurements is cancelled in the differencing operation [3, 22]. Also note that the bracketed coefficients in the equation above will be prominent in the \mathbf{H} matrix development discussed in Section 4.7.

In order to form the difference measurement, it is assumed above that the INS and transponder coordinates are expressed in the same frame. However, this *assumption* and *reality* are at odds. The RRS transponder whole-valued quantities are generally expressed in the ECEF frame (as defined by Britting). A simple transformation (presented in Chapter

It is used to express the transponder positions in the Litton ECEF frame. Litton [20] expresses the whole-valued INS position vector as a triplet [latitude, longitude, altitude] which can be converted to the (Litton) ECEF frame by [11]:

$$\begin{Bmatrix} x_U \\ y_U \\ z_U \end{Bmatrix} = \begin{Bmatrix} (RN + h) \cos \phi \sin \lambda \\ [(RN)(1 - e^2) + h] \sin \phi \\ (RN + h) \cos \phi \cos \lambda \end{Bmatrix} \quad (4.9)$$

where

ϕ	=	geodetic latitude
λ	=	terrestrial longitude
h	=	altitude above the reference ellipsoid
RN	=	$a_E / \sqrt{1 - e^2 \sin^2 \phi}$
a_E	=	earth's equatorial radius
e	=	eccentricity of earth's ellipsoid

Although both position vectors are now (apparently) expressed in the same frame, Equation (4.9) depends explicitly on the fact that the user position is known in terms of latitude, longitude, and altitude. The discussion in Chapter III presents the LN-93 error-angle vector $[\delta\theta_x, \delta\theta_y, \delta\theta_z]$ which may be transformed into latitude, longitude, and altitude. The approach is presented in detail in Section 4.6.

4.4 RRS Error-State Model Equations

The RRS error state vector is composed of 26 elements (shown in Table A.5, Appendix A). The RRS states occupy numbers $S_{INS} + 1$ through $S_{INS} + 26$ in the NRS error state model, where S_{INS} represents the total number of states used to model the INS subsystem in NRS. (The variation of S_{INS} is discussed in Chapter 6.) The first two RRS states are simple random constant (bias) states which model the effects of user hardware [RRS interrogator] range and range-rate calibration errors, respectively. Coupling of these states into the RRS measurement equation is presented in Sections 4.5 and 4.7. The error

state model equation for these states is [31]:

$$\begin{Bmatrix} \dot{x}_{br} \\ \dot{x}_{bv} \end{Bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{Bmatrix} x_{br} \\ x_{bv} \end{Bmatrix} \quad (4.10)$$

where

x_{br} = range equivalent of interrogator bias
 x_{bv} = velocity equivalent of interrogator bias

The initial state estimates and covariances for these states are [31]:

$$\begin{Bmatrix} \hat{x}_{br}(t_0) \\ \hat{x}_{bv}(t_0) \end{Bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (4.11)$$

and

$$P_{br,bv}(t_0) = \begin{bmatrix} 1ft^2 & 0 \\ 0 & 10^{-4}ft^2/sec^2 \end{bmatrix} \quad (4.12)$$

While the two states discussed above apply to all RRS measurements, there exist two sources of errors which are unique to each individual transponder. First is the error due to R^3 transponder surveyed position uncertainty (x, y, z components in ECEF frame), and second is the error due to atmospheric propagation delays between the user and each individual transponder. The three position error sources are well modeled by random bias states (as discussed in Chapter III), and the atmospheric error states are represented by first order Markov processes (also discussed in Chapter III). Then for each transponder, four states are used to define error sources [31]:

$$\begin{Bmatrix} \dot{x}_i \\ \dot{y}_i \\ \dot{z}_i \\ \dot{\delta R_{atm_i}} \end{Bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\frac{1}{\tau} \end{bmatrix} \begin{Bmatrix} x_i \\ y_i \\ z_i \\ \delta R_{atm_i} \end{Bmatrix} + \begin{Bmatrix} w_{x_i} \\ w_{y_i} \\ w_{z_i} \\ w_{atm_i} \end{Bmatrix} \quad (4.13)$$

The subscript i in the equation above denotes the i^{th} RRS transponder and $\tau = 300 \text{ sec}$ is the RRS transponder atmospheric error state time constant. The initial conditions for these states are [31];

$$\hat{\mathbf{x}}_{x,y,z,atm}(t_0) = \mathbf{0} \quad (4.14)$$

$$\mathbf{P}_{x,y,z,atm}(t_0) = \begin{bmatrix} 25ft^2 & 0 & 0 & 0 \\ 0 & 25ft^2 & 0 & 0 \\ 0 & 0 & 25ft^2 & 0 \\ 0 & 0 & 0 & 100(PPM)^2 \end{bmatrix} \quad (4.15)$$

and

$$E\{\mathbf{w}_{x,y,z,atm}\} = \mathbf{0} \quad (4.16)$$

$$E\{\mathbf{w}_{x,y,z,atm}(t)\mathbf{w}_{x,y,z,atm}(t+\tau)\} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{2\sigma_{atm}^2}{300} \end{bmatrix} \delta(\tau) \quad (4.17)$$

with $\sigma_{atm}^2 = 10^{-10}$. Once again, the set of equations above apply to a single transponder. There are six such sets of equations for RRS transponders which are used in this thesis. The error vector is specified in Appendix A.

4.5 Predicted RRS Measurement Equation

The Kalman filter combines range measurements $[\tilde{\mathbf{z}}]$ that are generated in the truth model with its own estimate of the measurements $[\hat{\mathbf{z}}]$ to calculate an optimal estimate of the state vector. The Kalman filter update equation which makes use of the actual and estimated measurements is:

$$\hat{\mathbf{x}}^+ = \hat{\mathbf{x}}^- + \mathbf{K}(\tilde{\mathbf{z}} - \hat{\mathbf{z}}) \quad (4.18)$$

where

$\hat{\mathbf{x}}^+$	=	filter estimate of state vector just after measurement
$\hat{\mathbf{x}}^-$	=	filter estimate of state just before measurement
\mathbf{K}	=	Kalman filter gain matrix

\tilde{z} = incoming measurement vector
 \hat{z}^- = Kalman filter prediction of incoming measurement vector

When measurements are processed sequentially, the measurement vector \tilde{z} becomes a series of scalar range measurements \tilde{z} . The Kalman filter prediction of each incoming measurement is also in the form of sequential scalar ranges measurements \hat{z} . The indicated positions of the transponder and user are modeled by:

$$\tilde{x}_T = x_T + \delta x_T \quad (4.19)$$

$$\tilde{x}_U = x_U + \delta x_U \quad (4.20)$$

where

\tilde{x}_T = surveyed transponder position vector
 x_T = true transponder position vector
 δx_T = true error in surveyed transponder position
 \tilde{x}_U = indicated user position vector (LN-93)
 x_U = true user position vector
 δx_U = true error in indicated user position

Subtracting the Kalman filter position error estimate $\hat{\delta x}_T$ and $\hat{\delta x}_U$ from the applicable equations yields the filter's best estimates of the transponder and user positions:

$$\hat{x}_T = \tilde{x}_T - \hat{\delta x}_T \quad (4.21)$$

$$= x_T + \delta x_T - \hat{\delta x}_T \quad (4.22)$$

$$= x_T + \text{estimation error}_T \quad (4.23)$$

$$\hat{x}_U = \tilde{x}_U - \hat{\delta x}_U \quad (4.24)$$

$$= x_U + \delta x_U - \hat{\delta x}_U \quad (4.25)$$

$$= x_U + \text{estimation error}_U \quad (4.26)$$

The Kalman filter estimate of the upcoming measurement is given by:

$$\widehat{\delta z} = \widehat{R}_{INS} - \widehat{R}_T \quad (4.27)$$

or

$$\widehat{\delta z} = \sqrt{(\widehat{x}_U - \widehat{x}_T)^2 + (\widehat{y}_U - \widehat{y}_T)^2 + (\widehat{z}_U - \widehat{z}_T)^2} - \delta \widehat{R}_{atm} - \delta \widehat{R}_{br} \quad (4.28)$$

where

$$\begin{aligned} \widehat{x}_T, \widehat{y}_T, \widehat{z}_T &= \text{filter estimate of transponder position} \\ \widehat{x}_U, \widehat{y}_U, \widehat{z}_U &= \text{filter estimate of user position} \end{aligned}$$

4.6 Transformation of User Position Error

In Section 4.3 it is assumed that user position and position errors are known in the ECEF frame. This section describes the transformations needed to process measurements in the ECEF frame and update the INS position error states in error-angle space (the LN-93 error-model space). The LN-93 Truth Model and Error Budget document [20] states that *globally valid* equations which transform $\delta\theta_x$, $\delta\theta_y$, and $\delta\theta_z$ to $\delta\phi$, $\delta\lambda$, and $\delta\alpha$ are generally quite complicated and seldom needed. For *non-polar* regions, the following approximate transformation is used:

$$\begin{Bmatrix} \delta\phi \\ \delta\lambda \\ \delta\alpha \end{Bmatrix} = \begin{Bmatrix} \delta\theta_y \sin \alpha - \delta\theta_x \cos \alpha \\ (\delta\theta_y \cos \alpha + \delta\theta_x \sin \alpha) \sec \phi \\ \delta\theta_z - \delta\lambda \sin \phi \end{Bmatrix} \quad (4.29)$$

$$\delta h = \delta h \quad (4.30)$$

The equations above may be expressed as a single matrix equation:

$$\begin{Bmatrix} \delta\phi \\ \delta\lambda \\ \delta\alpha \\ \delta h \end{Bmatrix} = \begin{bmatrix} -\cos\alpha & \sin\alpha & 0 & 0 \\ \sin\alpha \sec\phi & \cos\alpha \sec\phi & 0 & 0 \\ -\sin\alpha \tan\phi & -\cos\alpha \tan\phi & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} \delta\theta_w \\ \delta\theta_y \\ \delta\theta_z \\ \delta h \end{Bmatrix} = [C_u^n] \begin{Bmatrix} \delta\theta_x \\ \delta\theta_y \\ \delta\theta_z \\ \delta h \end{Bmatrix} \quad (4.31)$$

where C_u^n is the error-angle space to navigation error space transformation matrix defined in Chapter II. If user position is expressed in the ECEF frame as:

$$\mathbf{x}_u' = f(\phi, \lambda, \alpha, h) = [C_w^n] \begin{Bmatrix} 0 \\ 0 \\ R_E + h \end{Bmatrix} = \begin{Bmatrix} (R_E + h) \cos\phi \sin\lambda \\ (R_E + h) \sin\phi \\ (R_E + h) \cos\phi \cos\lambda \end{Bmatrix} \quad (4.32)$$

(where C_w^n is as yet undefined, but is intended to transform wander azimuth vectors to navigation frame vectors), then user position may also be expressed as:

$$\mathbf{x}_u'' = f(x, y, z) = \begin{Bmatrix} x'' \\ y'' \\ z'' \end{Bmatrix} \quad (4.33)$$

Equations (4.32) and (4.33) may be expanded in a Taylor series. Because this is a linearized error model, it is assumed that the higher order terms (h.o.t.'s) are negligible [3, 10, 22]. Then the series becomes $\mathbf{x}_u' = \mathbf{x}_u + \delta\mathbf{x}_u$, and the error vector term may be written as:

$$\begin{Bmatrix} \delta x'' \\ \delta y'' \\ \delta z'' \end{Bmatrix} = \frac{\partial f(\phi, \lambda, \alpha, h)}{\partial \mathbf{X}_u} \bigg|_{\mathbf{X}_{u, nom}} \begin{Bmatrix} \delta\phi \\ \delta\lambda \\ \delta\alpha \\ \delta h \end{Bmatrix} \quad (4.34)$$

where the user position vector is expressed in terms of latitude, longitude, wander angle, and altitude as:

$$\mathbf{X}_u = \left\{ \phi, \lambda, \alpha, h \right\}^T \quad (4.35)$$

and

$$\frac{\partial f(\phi, \lambda, \alpha, h)}{\partial \mathbf{X}_i} = \begin{bmatrix} -(R_E + h) \sin \phi \sin \lambda & (R_E + h) \cos \phi \cos \lambda & 0 & \cos \phi \sin \lambda \\ (R_E + h) \cos \phi & 0 & 0 & \sin \phi \\ -(R_E + h) \sin \phi \cos \lambda & -(R_E + h) \cos \phi \sin \lambda & 0 & \cos \phi \cos \lambda \end{bmatrix} \quad (4.36)$$

Equations (4.31), (4.34), and (4.36) are combined to obtain [11]:

$$\begin{Bmatrix} \delta x^u \\ \delta y^u \\ \delta z^u \end{Bmatrix} = \begin{bmatrix} L \end{bmatrix} \begin{Bmatrix} (R_E + h) \delta \theta_x \\ (R_E + h) \delta \theta_y \\ \delta h \end{Bmatrix} \quad (4.37)$$

where

$$\begin{aligned} L_{11} &= \sin \alpha \cos \lambda + \cos \alpha \sin \lambda \sin \phi \\ L_{12} &= \cos \alpha \cos \lambda - \sin \alpha \sin \lambda \sin \phi \\ L_{13} &= \sin \lambda \cos \phi \\ L_{21} &= -\cos \alpha \cos \phi \\ L_{22} &= \sin \alpha \cos \phi \\ L_{23} &= \sin \phi \\ L_{31} &= -\sin \alpha \sin \lambda + \cos \alpha \cos \lambda \sin \phi \\ L_{32} &= -\cos \alpha \sin \lambda - \sin \alpha \cos \lambda \sin \phi \\ L_{33} &= \cos \lambda \cos \phi \end{aligned}$$

Rearranging the the right-hand side of Equation (4.37) results in:

$$\begin{Bmatrix} \delta x^u \\ \delta y^u \\ \delta z^u \end{Bmatrix} = \begin{bmatrix} C_{uv}^u \end{bmatrix} \begin{Bmatrix} (R_E + h) \delta \theta_y \\ -(R_E + h) \delta \theta_x \\ \delta h \end{Bmatrix} \quad (4.38)$$

The matrix $[C_w^n]$ is the transformation matrix which converts a vector from the wander azimuth frame to the ECEF frame, and is written as:

$$\begin{Bmatrix} X_U \\ Y_U \\ Z_U \end{Bmatrix}^u = [C_w^n] \begin{Bmatrix} X_U \\ Y_U \\ Z_U \end{Bmatrix}^{waz} \quad (4.39)$$

The individual elements of C_w^n are given as [20]

$$\begin{aligned} C_{11} &= \cos \alpha \cos \lambda - \sin \alpha \sin \lambda \sin \phi \\ C_{12} &= -\sin \alpha \cos \lambda - \cos \alpha \sin \lambda \sin \phi \\ C_{13} &= \sin \lambda \cos \phi \\ C_{21} &= \sin \alpha \cos \phi \\ C_{22} &= \cos \alpha \cos \phi \\ C_{23} &= \sin \phi \\ C_{31} &= -\sin \lambda \cos \alpha - \cos \lambda \sin \phi \sin \alpha \\ C_{32} &= \sin \lambda \sin \alpha - \cos \lambda \sin \phi \cos \alpha \\ C_{33} &= \cos \lambda \cos \phi \end{aligned}$$

Then the filter estimate of the range between the user and the ground transponder is:

$$\hat{z} = \left| (\mathbf{x}_T + \delta \mathbf{x}_T - \hat{\delta \mathbf{x}}_T) - (\mathbf{x}_U + [C_w^n] \delta \mathbf{x}'_{U'} - [C_w^n] \hat{\delta \mathbf{x}}'_{U'}) \right| - \hat{\delta R}_{atm} - \hat{\delta R}_{br} \quad (4.40)$$

$$= |\mathbf{R}_t + \Sigma \mathbf{R}_{pos}| + \hat{\delta R}_{atm} + \hat{\delta R}_{br} \quad (4.41)$$

where:

$$\delta \mathbf{x}'_{U'} = \begin{Bmatrix} (R_E + h) \delta \theta_y \\ -(R_E + h) \delta \theta_x \\ \delta h \end{Bmatrix}, \quad \text{and} \quad \hat{\delta \mathbf{x}}'_{U'} = \begin{Bmatrix} (\hat{R}_E + \hat{h}) \hat{\delta \theta}_y \\ -(\hat{R}_E + \hat{h}) \hat{\delta \theta}_x \\ \hat{\delta h} \end{Bmatrix} \quad (4.42)$$

In the truth model, the measurement is given by:

$$\mathbf{z} = |\mathbf{x}_T - \mathbf{x}_U| - \delta R_{atm} - \delta R_{br} \quad (4.43)$$

$$= |\mathbf{R}_t| + \delta R_{atm} + \delta R_{br} \quad (4.44)$$

$$(4.45)$$

Then the residual, formed as the difference between Equation (4.44) and (4.41), results in the true whole range magnitude being cancelled:

$$\Delta z = z - \hat{z} \quad (4.46)$$

where z and \hat{z} are specified in Equations (4.44) and (4.41). It is the "new information" contained in the scalar residual which is scaled by time-varying Kalman filter gains (discussed in the next section) to update the filter state estimates.

4.7 Kalman Filter Gain

The Kalman filter gain matrix $K(t)$ used in Equation (4.18) determines which of the elements in the filter state vector are affected by measurements that are processed by the filter. The magnitude of the "correction" to individual states at update time is determined by the magnitude of the elements in the $K(t)$ matrix. The $K(t)$ matrix is computed prior to measurements from the following equation:

$$K(t_i) = P(t_i^-)H^T(t_i)[H(t_i)P(t_i^-)H^T(t_i) + R(t_i)]^{-1} \quad (4.47)$$

where the H matrix is the Jacobian gradient of the noiseless measurement vector [3] and the P matrix is the filter-computed covariance at time (t_i^-) . In the case of the RRS range measurements, six transponders are used to form measurements. For the purpose of software efficiency, the six measurements are assumed to occur cyclically at one second intervals (as opposed to actual hardware measurements which may occur asynchronously). Thus, a complete RRS measurement cycle takes six seconds, with one transponder measurement occurring every second.

The elements of the H matrix have previously been derived in terms of the ECEF frame measurement equation. However, in order to update the INS error states, the coefficients which precede the δX_U terms must be transformed back into the error-angle space in which INS error model computations take place. Note that the difference measurement

equation may be written in its functional form which is [11]:

$$\tilde{\delta z} = h(x_U, y_U, z_U, x_T, y_T, z_T, \delta R_{atm}, \delta R_{br}) + v \quad (4.48)$$

If the expression above is written as a Taylor series in which the terms above first order are neglected [3, 22], the result is:

$$\tilde{\delta z} \approx h(\cdot) \Big|_{(\cdot)_{nom}} + H(\cdot) \Big|_{(\cdot)_{nom}} \delta x \quad (4.49)$$

where:

$$H(\cdot) = \frac{\partial h(\cdot)}{\partial x} \quad (4.50)$$

and

$$(\cdot) = (x_U, y_U, z_U, x_T, y_T, z_T, \delta R_{atm}, \delta R_{br}) \quad (4.51)$$

The noiseless part of $\tilde{\delta z}$ is the original h function. The second term in Equation (4.49) is evaluated by taking eight partial derivatives as indicated by Equation (4.50). Only the non-zero partial derivatives indicated by Equation (4.50) are discussed below; most of the elements of the corresponding H row will also be zero [11] due to the fact that the measurement depends upon only eight states [represented by the arguments contained in (\cdot) above]. The non-zero elements of H are shown in the following row vector:

$$\left[\frac{\partial h}{\partial x_U} \quad \frac{\partial h}{\partial y_U} \quad \frac{\partial h}{\partial z_U} \quad \frac{\partial h}{\partial x_T} \quad \frac{\partial h}{\partial y_T} \quad \frac{\partial h}{\partial z_T} \quad \frac{\partial h}{\partial \delta R_{atm}} \quad \frac{\partial h}{\partial \delta R_{br}} \right]$$

where the first three terms (after transformation back into wander azimuth coordinates) will affect (INS) states 1, 2, and 10. The δR_{br} term is the RRS "common" bias state which affects all transponder measurements and represents the $H(S_{INS} + 1)$ element. The remaining four terms are specific to the transponder which is being "interrogated" and affect the respective x, y, z , and atm error states in the RRS model. As an example, for the 98-state model presented in Chapter VI and Appendix F, these four terms are $H(75)$, $H(76)$, $H(77)$, and $H(78)$ for a measurement from RRS transponder number one.

Noting that the $h(\cdot)$ function is linear in the error terms due to equipment induced bias and the atmospheric error, the applicable row of the H matrix may be written:

$$\left[\frac{\partial h}{\partial x_U} \quad \frac{\partial h}{\partial y_U} \quad \frac{\partial h}{\partial z_U} \quad \frac{\partial h}{\partial x_T} \quad \frac{\partial h}{\partial y_T} \quad \frac{\partial h}{\partial z_T} \quad 1 \quad 1 \right]$$

Evaluating the partial derivatives of $h(\cdot)$ with respect to the transponder position error states reveals that these terms are simply the components of the unit line-of-sight from the user to the transponder, expressed in the ECEF frame [11]. The H row may be written as:

$$\left[\frac{\partial h}{\partial x_U} \quad \frac{\partial h}{\partial y_U} \quad \frac{\partial h}{\partial z_U} \quad ULOS_x \quad ULOS_y \quad ULOS_z \quad 1 \quad 1 \right]$$

The following section is adapted from a development in [11]. Referring to Equation (4.49), the first order term (evaluated with respect to the user position error states *only*) may be expressed in the ECEF frame as:

$$H(\cdot) \Big|_{(\cdot)_{nom}} \delta x = \frac{\partial h}{\partial \mathbf{x}_U'} \Big|_{nom} \delta \mathbf{x}_{U'}'' \quad (4.52)$$

Again, the arguments represented by (\cdot) in the equation above are now limited to the user position states $[x_U, y_U, z_U]$. A 3-by-3 identity matrix may be inserted between the "partial" term and the error vector without changing the result:

$$H(\cdot) \Big|_{(\cdot)_{nom}} \delta x = \frac{\partial h}{\partial \mathbf{x}_U'} \Big|_{nom} \mathbf{I} \delta \mathbf{x}_{U'}'' \quad (4.53)$$

Now replace the identity matrix by two direction cosine matrices [which when multiplied together would return the identity matrix]:

$$H(\cdot) \Big|_{(\cdot)_{nom}} \delta x = \frac{\partial h}{\partial \mathbf{x}_U'} \Big|_{nom} \mathbf{C}_w' \mathbf{C}_t'' \delta \mathbf{x}_{U'}'' \quad (4.54)$$

Evaluating the partial derivative and multiplying out the product $\mathbf{C}_t'' \delta \mathbf{x}_{U'}''$ leaves:

$$H(\cdot) \Big|_{(\cdot)_{nom}} \delta x = \left[-ULOS_x \quad -ULOS_y \quad -ULOS_z \right] \Big|_{nom} \mathbf{C}_w' \delta \mathbf{x}_{U'}'' \quad (4.55)$$

If the row vector and the DCM are multiplied together, a row vector is obtained. For convenience, the result is defined as $[a \ b \ c]$. The right half of the product can be rewritten using the results of Equation (4.42):

$$H(\cdot) \Big|_{(\cdot)_{num}} \delta x = [a \ b \ c] \cdot \begin{Bmatrix} (R_E + h)\delta\theta_y \\ -(R_E + h)\delta\theta_x \\ \delta h \end{Bmatrix} \quad (4.56)$$

Finally, the expression above may be rewritten to yield:

$$H(\cdot) \Big|_{(\cdot)_{num}} \delta x = [-b(R_E + h) \ a(R_E + h) \ c] \cdot \begin{Bmatrix} \delta\theta_x \\ \delta\theta_y \\ \delta h \end{Bmatrix} \quad (4.57)$$

Then the first three **H** elements in the row are $-b(R_E + h)$, $a(R_E + h)$, and c . Equations for all other non-zero elements in **H** are developed above. These equations are programmed into MSOFE for the RRS measurement model.

4.8 Transponder Physical Locations

The six transponders which are modeled in the NRS composite error model are those which are on and near Holloman AFB, NM. The locations of these transponders have been surveyed (typically to within 5 feet in each of the 3 axes), and the coordinates are given in Table 4.1.

Table 4.1. RRS Transponder Locations [31]

Transponder ID	Latitude	Longitude	Altitude
005	33° 01' 36.1472"	- 106° 08' 20.7404"	4339 ft
102	32° 55' 58.5986"	- 106° 08' 50.3339"	4074 ft
181	33° 44' 58.035"	- 106° 22' 14.630"	7932 ft
211	33° 17' 55.999"	- 106° 31' 44.311"	8842 ft
212	32° 47' 16.418"	- 105° 49' 15.474"	9202 ft
216	32° 42' 12.235"	- 106° 07' 38.907"	4481 ft

4.9 Summary

This chapter presents the basic concepts related to the RRS transponder system and includes the RRS transponder system error model equations as well. In addition, the range measurement model equation is developed in detail. The conventional *difference* measurement approach is presented and discussed, chiefly to point out that the method results in the whole-valued quantity being canceled in the "delta-measurement". Finally, the calculations required to generate H matrix elements for a typical RRS range measurement are demonstrated.

V. Global Positioning System Model and Space Vehicle Orbit Calculations

5.1 Introduction

The Global Positioning System (GPS) is designed to be a highly accurate, stand-alone navigation system. However, for this research, GPS is used as a subsystem to improve the navigation solution of the LN-93 based NRS. In a manner somewhat reminiscent of the RRS transponder system discussed in Chapter IV, GPS navigation information is obtained from electromagnetic signal propagation through the media (space and atmosphere) between the user (NRS) and each of the space vehicles (SVs) which the user "locks" into a reception channel of the GPS receiver. Navigation information is obtained by receiving GPS SV ephemeris data which are broadcast continuously from each active ("locked-on") SV, correlating the phase of the signal with a matching signal in the GPS receiver, and correcting for known error sources to produce a highly accurate range estimate between the user and each SV which is monitored. Although not used in this thesis, range-rate information may be obtained from GPS ephemeris in a similar manner. As in RRS, GPS range (and, if used, range-rate) measurements make refinements to the NRS navigation solution possible.

Figure 5.1 depicts the addition of the GPS subsystem to the RRS and baro-aided INS diagram presented in Chapter IV. This is the final module addition and completes the full NRS model.

A dynamic error model for the GPS system is developed in the Solomon thesis [33] and revised in [11]. Portions of those bodies of work are summarized in this chapter. However, *substantial changes* to the basic GPS model are made. In the references cited [11, 33], a simplified GPS model was assumed. It consisted of four *stationary* space vehicles (SVs) and did not perform geometry optimization calculations. In this thesis, a 24-SV "optimal" constellation based on a paper by Green [12] is modeled. The new model includes orbital calculations for all SVs, and *simulates GPS receiver operation* as well. These enhancements are discussed in detail in subsequent sections of this chapter. As in the case of RRS measurements, the lever-arm effect is important to consider in actual GPS hardware applications. Once again, the lever-arm effect is avoided by assuming that the

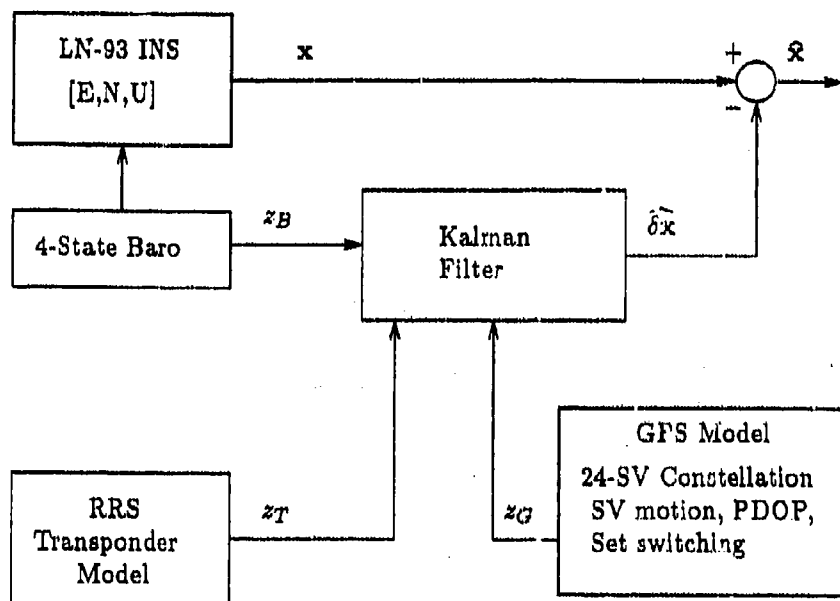


Figure 5.1. NRS: LN-93 INS Aided by Baro-Altimeter, RRS, and GPS

INS and the GPS antenna are collocated. The assumption of collocation is made in the interest of generality and NRS software efficiency. The *measurement* model equations for the GPS system follow a parallel development to that of the RRS measurement model in Chapter IV.

5.2 GPS Range Measurements

Once again, in ARS or the proposed NRS, GPS range measurements aid in estimating position errors of the reference system. The GPS range measurement is derived from decoding ephemeris data which are broadcast continuously by each active SV. The user's GPS receiver (now considered to be a subsystem in NRS) processes signals which are received from the GPS SVs to determine pseudo-range between the user and the SV. The range measurement thus obtained is corrupted by several error sources which must be determined and compensated.

In its simplest form, a range measurement between a single GPS SV and the user (in this case, NRS) may be determined as the product of propagation speed of the electro-

magnetic (EM) signal and elapsed time during such propagation. Stated mathematically, the range (typically called pseudo-range due to inherent inaccuracies in the estimate) is given by:

$$R_p = c \cdot t_e \quad (5.1)$$

where R_p is pseudo-range, c is the speed of light in vacuum, and t_e represents the elapsed time for transit of the EM signal.

However, two major problems exist in using this simple definition for range. First, the EM signal is not propagated entirely in vacuum. The signal originates in space (where the assumption of vacuum is acceptable), but must subsequently propagate some distance through the earth's atmosphere as well. Naturally, the *signal delay* introduced by atmospheric propagation must be taken into account. Second, in order to preserve any hope of accurately determining range, it is *critical* to determine the EM signal transit time with an extremely high degree of accuracy. Recalling that light (or any EM signal) propagates on the order of $3 \times 10^8 m/s$, it is readily apparent that even a very small error in determining the EM signal propagation time can wreak havoc on attempts to use pseudo-range information to improve the navigation system solution in NRS (or any other such system).

As a consequence of the concerns above, it is imperative to develop a much higher fidelity model for range estimation. A typical GPS receiver models the range between the user and space vehicles with the following equation [32]:

$$R_{GPS} = R_t + \delta R_{cl} + \delta R_{trop} + \delta R_{ion} + \delta R_{svclk} + \delta R_{uclk} + v \quad (5.2)$$

where

R_{GPS}	=	GPS pseudo-range measurement, from SV to user
R_t	=	True range, from SV to user
δR_{cl}	=	Range error due to code loop error
δR_{trop}	=	Range error due to tropospheric delay
δR_{ion}	=	Range error due to ionospheric delay
δR_{svclk}	=	Range error due to SV clock error
δR_{uclk}	=	Range error due to User clock error
v	=	zero-mean white Gaussian measurement noise

As in the comparable equation for RRS, the GPS pseudo-range equation above includes the true range [which can never be known exactly] along with terms which reflect sources of error and uncertainty inherent to GPS range measurements. GPS error sources and models are discussed later.

5.3 GPS Pseudo-Range Calculation From INS Data

As in the RRS subsystem, it is desirable to formulate a *difference* measurement in the GPS model as well. Once again, two sources of range information must be obtained. Like the RRS case presented in Chapter IV, the first source is the range *measurement* which comes from the GPS subsystem and which is *modeled* by Equation (5.1). The second range estimate is constructed by differencing INS-indicated position and SV (broadcast) positions to calculate the range. (Note that the indicated INS position and SV position contain uncertainties which must be considered.) In Chapter IV, the user (INS) position is represented by an R^3 vector expressed in the Litton ECEF which is repeated below for convenience. SV position, \mathbf{X}_s , is represented in like fashion.

$$\mathbf{X}_U = \begin{Bmatrix} x_U \\ y_U \\ z_U \end{Bmatrix}^T, \quad \mathbf{X}_s = \begin{Bmatrix} x_s \\ y_s \\ z_s \end{Bmatrix}^T \quad (5.3)$$

Then user to SV pseudo-range may be calculated as:

$$R_{INS} = \left| \mathbf{X}_U - \mathbf{X}_s \right| = \left| \begin{Bmatrix} x_U \\ y_U \\ z_U \end{Bmatrix}^T - \begin{Bmatrix} x_s \\ y_s \\ z_s \end{Bmatrix}^T \right| \quad (5.4)$$

Equation (5.4) may also be written in the equivalent form:

$$R_{INS} = \sqrt{(x_U - x_s)^2 + (y_U - y_s)^2 + (z_U - z_s)^2} \quad (5.5)$$

Invoking perturbation theory [3, 22], the equation above is written as a first-order Taylor series to approximate the (INS derived) calculation of user to SV range. The truncated (to first-order) series is expressed as:

$$R_{INS} = R_t + \left. \frac{\partial R_{INS}(\mathbf{X}_S, \mathbf{X}_U)}{\partial \mathbf{X}_S} \right|_{(\mathbf{X}_S, \mathbf{X}_U)_{nom}} \cdot \delta \mathbf{X}_S + \left. \frac{\partial R_{INS}(\mathbf{X}_S, \mathbf{X}_U)}{\partial \mathbf{X}_U} \right|_{(\mathbf{X}_S, \mathbf{X}_U)_{nom}} \cdot \delta \mathbf{X}_U \quad (5.6)$$

When equation (5.5) is substituted into Equation (5.6) and the partial derivatives evaluated, the INS-derived pseudo-range approximation becomes:

$$R_{INS} = R_t - \left[\frac{x_S - x_U}{|R_{INS}|} \right] \cdot \delta x_U - \left[\frac{y_S - y_U}{|R_{INS}|} \right] \cdot \delta y_U - \left[\frac{z_S - z_U}{|R_{INS}|} \right] \cdot \delta z_U + \left[\frac{x_S - x_U}{|R_{INS}|} \right] \cdot \delta x_S + \left[\frac{y_S - y_U}{|R_{INS}|} \right] \cdot \delta y_S + \left[\frac{z_S - z_U}{|R_{INS}|} \right] \cdot \delta z_S \quad (5.7)$$

Now the GPS pseudo-range *difference* measurement is formed as:

$$\begin{aligned} \delta z &= R_{INS} - R_s \\ &= - \left[\frac{x_S - x_U}{|R_{INS}|} \right] \cdot \delta x_U - \left[\frac{y_S - y_U}{|R_{INS}|} \right] \cdot \delta y_U - \left[\frac{z_S - z_U}{|R_{INS}|} \right] \cdot \delta z_U \\ &\quad + \left[\frac{x_S - x_U}{|R_{INS}|} \right] \cdot \delta x_S + \left[\frac{y_S - y_U}{|R_{INS}|} \right] \cdot \delta y_S + \left[\frac{z_S - z_U}{|R_{INS}|} \right] \cdot \delta z_S \\ &\quad - [1] \delta R_{cl} - [1] \delta R_{trop} - [1] \delta R_{ion} - [1] \delta R_{setk} - [1] \delta R_{clock} + v \quad (5.8) \end{aligned}$$

As noted in the RRS case, the true whole-valued range (R_t) formerly present in the individual pseudo-range representations (R_{INS} and R_{GPS}) is cancelled in the differencing operation. The bracketed coefficients in the equation above will again be used in the \mathbf{H} matrix development discussed in a later section.

In order to form the GPS difference measurement, it is assumed above that the INS and SV coordinates are expressed in the same frame. However, the *assumption* and *reality*

are at odds once again. The SV whole-valued quantities are generally expressed in either their own orbital frame or possibly in the inertial frame (as defined by Britting [3]). In Chapter IV, a transformation is presented that takes the whole-valued INS position vector triplet [latitude, longitude, altitude] and converts it to the (Litton) ECEF frame as shown below:

$$\begin{Bmatrix} x_U \\ y_U \\ z_U \end{Bmatrix}^c = \begin{Bmatrix} (RN + h) \cos \phi \sin \lambda \\ [(RN)(1 - e^2) + h] \sin \phi \\ (RN + h) \cos \phi \cos \lambda \end{Bmatrix}^c \quad (5.9)$$

Equation (5.9) depends on knowing the user position in terms of latitude, longitude, and altitude. The discussion in Chapter III presents the LN-93 error-angle vector $[\delta\theta_x, \delta\theta_y, \delta\theta_z]$ which may be transformed into latitude, longitude, and altitude error space. The approach for obtaining the INS position error in ECEF coordinates (from the $[\delta L, \delta\lambda, \delta h]$ triplet) is presented in detail in Section 4.6.

The SV positions are routinely defined in terms of *orbital* parameters which may then be defined in terms of inertial space coordinates. Additionally, the "flight" of the SVs in their orbital planes and the effect of earth rotation must be taken into account. SV positions are (carefully) transformed to the Litton ECEF as discussed below.

Each of the 24 SVs is assigned an initial position (refer to Sections 5.8 and 5.9) based on optimization of global GPS coverage using the 24 SV constellation [12]. Four SVs are assigned to each of six orbital planes; each orbital plane is inclined at 55° with respect to the inertial space x_i, y_i plane. The six lines of nodes [also called longitudes of ascending nodes or LANs] are equally spaced around the circumference of the earth.

A single (representative) orbital plane is shown in Figure 5.3. The orthogonal axes $[\bar{I}, \bar{J}, \bar{K}]$ are identical to the $[x_i, y_i, z_i]$ axes defined in the Britting inertial frame discussed in Chapter II. The LAN is the point at which an SV in this orbit crosses the equator traveling from south to north, and is represented in the figure as \bar{n} . Additionally, the angle between the LAN and the inertial frame axis x_i is designated Ω . The angular momentum vector (\bar{h} in the figure) is normal to the orbital plane, and the angle between \bar{h} and \bar{K} is called the angle of inclination and is given the symbol β for this document. Besides the orbital

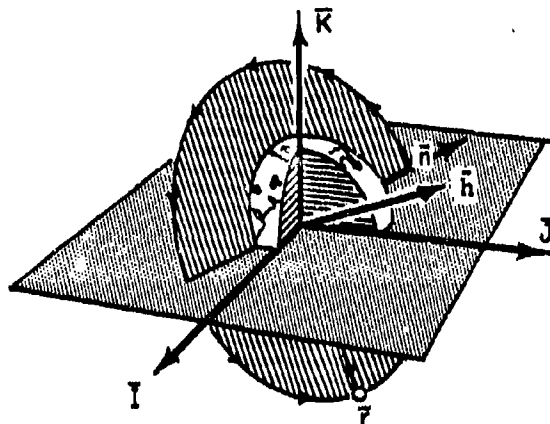


Figure 5.2. Orbital Path for a Typical SV

plane angle of inclination (β) and the LAN angular displacement from the x_i axis (Ω), it is also necessary to know the position of each SV in its orbital path at some point in time, say t_0 . If initial conditions are known, it is a straightforward calculation to determine SV position *in the orbital plane* at any time subsequent to t_0 .

Although the figure depicts elliptical orbits, the model assumed in this research is that of circular orbits. This assumption is based on the optimal SV constellation described in [12]. In the optimal constellation (which is currently being implemented in practice), SV orbital semi-major axes are greater (26,609 km) than previous constellations, which were on the order of 26,560 km. The net effects are (a) a more circular orbit, (b) improved coverage, and (c) reduced ellipticity in the orbital paths. While the ellipticity is tabularized as zero in the Green paper [12], the actual ellipticity is non-zero, but its effect is sufficiently small to be neglected for the purpose of this research. If later investigation proves to the contrary, then a change to the SV orbit calculations may be made to account for ellipticity. At this point the SV orbital model is developed.

Consider for a moment that each GPS SV travels in a (nearly) circular path in a single plane, irrespective of the orientation of that plane. In a situation such as this, the SV position is depicted in Figure 5.3 as a function of orbital radius and an angle α . The angle

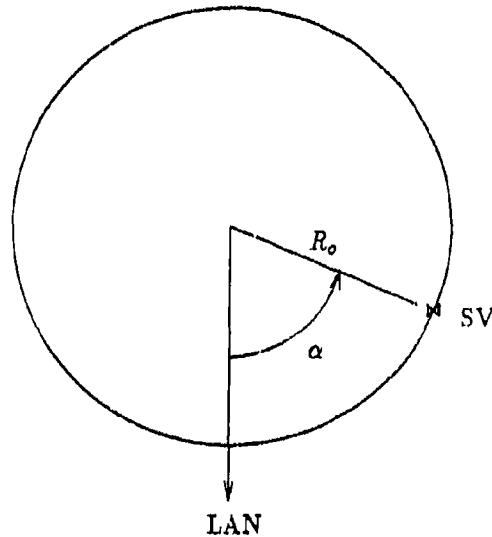


Figure 5.3. Planar Representation of SV Orbital Path

alpha is defined (conveniently) with respect to the LAN, and is described mathematically as:

$$\alpha(t) = \alpha_0 + \omega_n \cdot t \quad (5.10)$$

Now, define an orthogonal frame in R^3 which has its origin coincident with the orbital plane origin shown in Figure 5.3. The new frame x_0 axis is collinear with the orbital plane LAN; the new y_0 axis is 90 degrees counterclockwise from and coplanar with x_0 , and z_0 is pointing out of the page. The newly defined frame is shown in Figure 5.3. Note that the SV position as defined in this frame is still "two-dimensional," in that the z_0 component of the SV position vector is identically zero.

Keeping in mind that α is time-varying, the SV position in the orthogonal orbital frame depicted in Figure 5.3 is given as:

$$\begin{Bmatrix} x_0 \\ y_0 \\ z_0 \end{Bmatrix} = \begin{Bmatrix} R \cos \alpha \\ R \sin \alpha \\ 0 \end{Bmatrix} \quad (5.11)$$

Now it remains to rotate the SV position expressed in the orbital frame above into the inertial frame, taking care to account for the orbital plane angle of inclination and

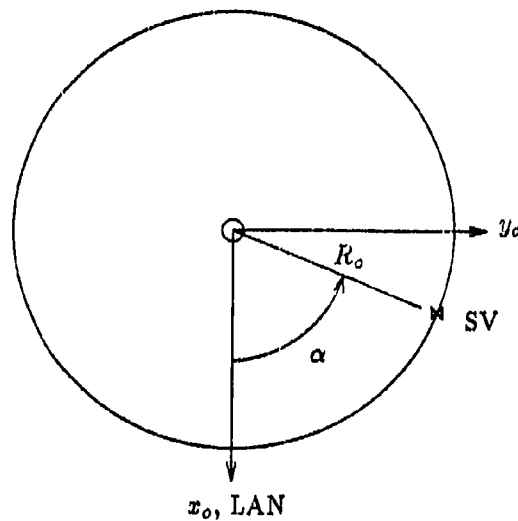


Figure 5.4. Orthogonal Orbital Frame Definition

the angular displacement of the LAN from the inertial frame x_i axis. Two rotations are needed to accomplish the transformation. First, the orbital frame is rotated clockwise about its own x_o axis through an angle of 55° (the orbital plane angle of inclination, β). Second, the orbital frame is rotated counterclockwise about the z_o axis through an angle equal to the original angle from the x_i axis to the line of ascending nodes (LAN), Ω . After these two rotations, the orthogonal orbital frame is coincident with the inertial frame. Thus, an SV position vector expressed in the orthogonal orbital frame is transformed to the inertial frame via:

$$\begin{Bmatrix} x_i \\ y_i \\ z_i \end{Bmatrix} = \begin{Bmatrix} \cos \Omega & -\cos \beta \sin \Omega & \sin \beta \cos \Omega \\ \sin \Omega & \sin \beta \cos \Omega & -\sin \beta \cos \Omega \\ 0 & \sin \beta & \cos \beta \end{Bmatrix} \begin{Bmatrix} x_o \\ y_o \\ z_o \end{Bmatrix} \quad (5.12)$$

where β is the orbital plan angle of inclination (55°), and Ω is previously defined. The result thus obtained is a step closer to getting SV position vectors in terms of ECEF coordinates. Two more steps must be taken. First a transform from inertial space to the

(Britting) ECEF is given by [3]:

$$C_i^e = \begin{Bmatrix} \cos(\omega_{ie}t) & \sin(\omega_{ie}t) & 0 \\ -\sin(\omega_{ie}t) & \cos(\omega_{ie}t) & 0 \\ 0 & 0 & 1 \end{Bmatrix} \quad (5.13)$$

where ω_{ie} is the earth's rotation rate and t is the elapsed time since t_0 . The same value for t is used here and in the defining equation for the SV orbital angle α . Finally, the difference between the Britting ECEF and Litton ECEF must be taken into account to complete the transform. Using the matrix specified in Chapter II, SV positions in Litton ECEF terms are given by:

$$\begin{Bmatrix} x_r \\ y_r \\ z_r \end{Bmatrix}_{Litton} = C_n^h \cdot C_i^e \cdot C_o^i \cdot \begin{Bmatrix} x_o \\ y_o \\ z_o \end{Bmatrix} \quad (5.14)$$

At last the SV position is expressed in the desired frame. However, recall that the INS position vector above is expressed in terms of latitude, longitude, and altitude. Once again, the INS position must be expressed in terms of its error-angle states but coordinatized in the Litton ECEF. The approach is identical to that explained in Chapter IV and is omitted here.

5.4 GPS Error-State Model Equations

The GPS error state vector is composed of 30 elements (shown in Table A.6, Appendix A). The GPS states occupy the thirty "uppermost" states in the NRS error state model. The first two GPS states model the user set [receiver] clock bias and drift errors, respectively. The error state model equation for these states is [11, 33]:

$$\begin{Bmatrix} \dot{x}_{clk_b} \\ \dot{x}_{clk_{dr}} \end{Bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{Bmatrix} x_{clk_b} \\ x_{clk_{dr}} \end{Bmatrix} \quad (5.15)$$

where

\hat{x}_{Uclk_b} = range equivalent of user set clock bias
 $\hat{x}_{Uclk_{dr}}$ = velocity equivalent of user set clock drift

The initial state estimates and covariances for these states are [31]:

$$\begin{Bmatrix} \hat{x}_{Uclk_b}(t_0) \\ \hat{x}_{Uclk_{dr}}(t_0) \end{Bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (5.16)$$

and

$$\mathbf{P}_{Uclk_b, Uclk_{dr}}(t_0) = \begin{bmatrix} 9.0 \times 10^{11} ft^2 & 0 \\ 0 & 9.0 \times 10^{10} ft^2/sec^2 \end{bmatrix} \quad (5.17)$$

Note the large uncertainties associated with the user clock states. Until the user clock error is determined, it is the single largest source of error in GPS range measurements. While the two states discussed above apply to all GPS measurements, there exist five sources of errors which are unique to each individual SV.

One error source specific to each SV is code loop range quantization error. At the heart of any GPS receiver exists a pair of interacting tracking loops [21]. One of these loops, the "code tracking loop" is the source of pseudo-range estimation error which is modeled as a first order Markov process [21] with an exponential autocorrelation function. Other significant error sources include the tropospheric and ionospheric propagation delays. Both of these error sources are identified and corrected to a large degree by the GPS receiver. However the uncompensated error contribution of these error sources is still significant. Both of these are also modeled as first order Markov processes (with different time constants). Still other sources of error which must be included in the GPS model are SV clock error and SV R^3 position error, each of which is treated as a random bias state. The reader should note the strong possibility for observability problems in a model such as this. In this case, three position error states are used, but the measurements provide new information *only along the line-of-sight vector* between the user and SV. All error sources discussed above which are unique to each SV are included in an error state vector shown

below.

$$\begin{Bmatrix} \delta \dot{R}_{cl} \\ \delta \dot{R}_{trop} \\ \delta \dot{R}_{ion} \\ \delta \dot{R}_{selk} \\ \delta \dot{x}_{s_i} \\ \delta \dot{y}_{s_i} \\ \delta \dot{z}_{s_i} \end{Bmatrix} = \begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -\frac{1}{800} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{1500} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{Bmatrix} \delta R_{cl} \\ \delta R_{trop} \\ \delta R_{ion} \\ \delta R_{selk} \\ \delta x_{s_i} \\ \delta y_{s_i} \\ \delta z_{s_i} \end{Bmatrix} + \begin{Bmatrix} w_{cl} \\ w_{trop} \\ w_{ion} \\ 0 \\ 0 \\ 0 \\ 0 \end{Bmatrix} \quad (5.18)$$

$$P_{GPS}(t_0) = \begin{bmatrix} 0.25 ft^2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1.0 ft^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1.0 ft^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 25 ft^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 25 ft^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 25 ft^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 25 ft^2 \end{bmatrix} \quad (5.19)$$

and

$$E\{w_{GPS}(t)\} = 0 \quad (5.20)$$

$$E\{w_{GPS}(t)w_{GPS}(t+\tau)\} = \begin{bmatrix} 0.5 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.004 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.004 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \delta(\tau) \quad (5.21)$$

Once again, the set of equations above apply to a single SV. There are four such sets of matrix equations for GPS SV errors modeled in this thesis. The error-state vector is completely specified in Appendix A.

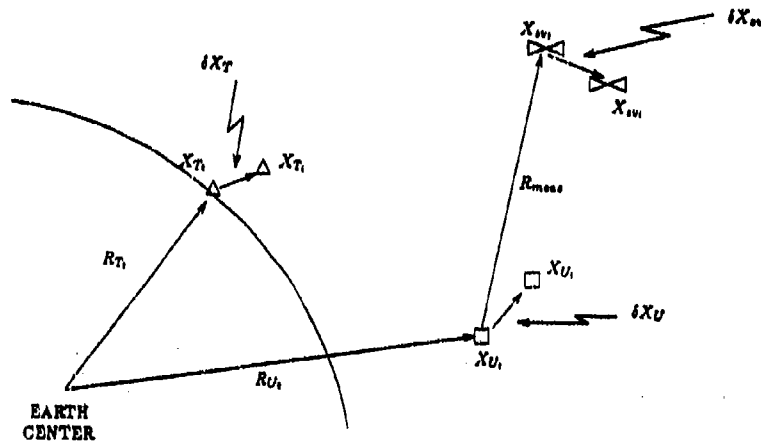


Figure 5.5. Two-Dimensional Representation of User and SV Position Errors.

5.5 Predicted GPS Measurement Equation

The Kalman filter combines range measurements $[\bar{z}]$ that are generated in the GPS truth model with its own prediction of the measurements $[\hat{z}^-]$ to calculate an optimal estimate of the state vector. The general Kalman filter update equation [22] defined in Chapter IV is repeated below:

$$\hat{\mathbf{x}}^+ = \hat{\mathbf{x}}^- + \mathbf{K}(\bar{\mathbf{z}} - \hat{\mathbf{z}}^-) \quad (5.22)$$

When GPS measurements are processed sequentially, the measurement vector $\bar{\mathbf{z}}$ becomes a series of scalar range measurements \bar{z} . The Kalman filter prediction of each incoming measurement is also in the form of sequential scalar ranges measurements \hat{z}^- . The indicated positions of the SV and user are depicted in two dimensions in Figure 5.5. The position errors depicted in Figure 5.5 are extended into R^3 space and are represented mathematically by:

$$\bar{\mathbf{x}}_s = \mathbf{x}_s + \delta \mathbf{x}_s \quad (5.23)$$

$$\bar{\mathbf{x}}_u = \mathbf{x}_u + \delta \mathbf{x}_u \quad (5.24)$$

where

$\bar{\mathbf{x}}_s$	=	indicated SV position vector
\mathbf{x}_s	=	true SV position vector
$\delta\mathbf{x}_s$	=	true error in indicated SV position
$\bar{\mathbf{x}}_U$	=	indicated user position vector (LN-93)
\mathbf{x}_U	=	true user position vector
$\delta\mathbf{x}_U$	=	true error in indicated user position

Subtracting the Kalman filter position error estimates $\widehat{\delta\mathbf{x}}_s^-$ and $\widehat{\delta\mathbf{x}}_U^-$ from the applicable equations yields the filter's best estimates of the SV and user positions:

$$\hat{\mathbf{x}}_s = \bar{\mathbf{x}}_s - \widehat{\delta\mathbf{x}}_s^- \quad (5.25)$$

$$= \mathbf{x}_s + \delta\mathbf{x}_s - \widehat{\delta\mathbf{x}}_s^- \quad (5.26)$$

$$= \mathbf{x}_s + error_s \quad (5.27)$$

$$\hat{\mathbf{x}}_U = \bar{\mathbf{x}}_U - \widehat{\delta\mathbf{x}}_U^- \quad (5.28)$$

$$= \mathbf{x}_U + \delta\mathbf{x}_U - \widehat{\delta\mathbf{x}}_U^- \quad (5.29)$$

$$= \mathbf{x}_U + error_U \quad (5.30)$$

The Kalman filter estimate of the upcoming measurement is given by:

$$\widehat{\delta z}^- = \hat{R}_{INS}^- - \hat{R}_s^- \quad (5.31)$$

or

$$\widehat{\delta z}^- = \sqrt{(\hat{x}_U - \hat{x}_s)^2 + (\hat{y}_U - \hat{y}_s)^2 + (\hat{z}_U - \hat{z}_s)^2} - \widehat{\delta R}_{cl} - \widehat{\delta R}_{trop} - \widehat{\delta R}_{ion} - \widehat{\delta R}_{Selk} \quad (5.32)$$

where

$\hat{x}_U, \hat{y}_U, \hat{z}_U$	=	filter estimate of user position
$\hat{x}_s, \hat{y}_s, \hat{z}_s$	=	filter estimate of SV position

5.6 GPS Measurement Estimation

In Section 5.3 it is assumed that user position and position errors are expressed in the ECEF frame. This section describes the transformations needed to process GPS measurements in the ECEF frame and update the INS position error states in error-angle space (the LN-93 error-model space). Using the exact same approach presented in Chapter IV, the INS position is written as:

$$\begin{Bmatrix} X_U \\ Y_U \\ Z_U \end{Bmatrix}^e = \begin{bmatrix} C_w^e \end{bmatrix} \begin{Bmatrix} X_U \\ Y_U \\ Z_U \end{Bmatrix}^{was} \quad (5.33)$$

Consequently, the filter estimate of range from user to SV may be expressed as:

$$\hat{z}^- = \left| (\mathbf{x}_s + \delta \mathbf{x}_s - \widehat{\delta \mathbf{x}}_s^-) - (\mathbf{x}_U + [C_w^e] \delta \mathbf{x}_U' - [C_w^e] \widehat{\delta \mathbf{x}}_U'^-) \right|$$

$$- \widehat{\delta R}_{cl}^- - \widehat{\delta R}_{trop}^- - \widehat{\delta R}_{ion}^- - \widehat{\delta R}_{Sclk}^- \quad (5.34)$$

$$(5.35)$$

where:

$$\delta \mathbf{x}_U' = \begin{Bmatrix} (R_E + h) \delta \theta_y \\ -(R_E + h) \delta \theta_x \\ \delta h \end{Bmatrix}, \quad \text{and} \quad \widehat{\delta \mathbf{x}}_U'^- = \begin{Bmatrix} (R_E + h) \widehat{\delta \theta}_y^- \\ -(R_E + h) \widehat{\delta \theta}_x^- \\ \widehat{\delta h}^- \end{Bmatrix} \quad (5.36)$$

In the truth model, the measurement is given by:

$$\mathbf{z} = |\mathbf{x}_U - \mathbf{x}_s| - \delta R_{cl} - \delta R_{trop} - \delta R_{ion} - \delta R_{Sclk} - \delta R_{Uclk} \quad (5.37)$$

$$(5.38)$$

In the residual below, the true whole range magnitudes are again cancelled:

$$\Delta \mathbf{z} = \mathbf{z} - \hat{\mathbf{z}} \quad (5.39)$$

It is the "new information" contained in the scalar residual which is scaled by time-varying Kalman filter gains (discussed in the next section) to update the filter state estimates.

5.7 GPS Measurement Matrix

The observation matrix $H(t)$ elements associated with the GPS measurements must be obtained in order to complete the Kalman filter update process. Four GPS measurements are assumed to occur simultaneously at 0.25Hz. Thus, a complete GPS measurement cycle takes four seconds, with all four SV measurements being processed sequentially at update times.

The GPS elements of the H matrix have previously been derived in terms of the ECEF frame measurement equation. As demonstrated in Chapter IV, the coefficients which precede the terms in the measurement difference equation are in fact the elements of the H row for the measurement being processed. Referring to Equation (5.8), the first three bracketed coefficients map into the user position states, the next three coefficients map into the SV position states, and the last four (ones) map into the remaining GPS error states.

For example, if SV_1 is used to form a GPS range measurement, coefficients 1, 2, and 3 in Equation (5.8) become elements 1, 2, and 10 in the H row. These element numbers correspond to the R^3 position errors for the user INS. Similarly, coefficients 4, 5, and 6 in Equation (5.8) become elements 126, 127, and 128 (corresponding to the position error states for SV number four). The remaining (unity) coefficients map into the remaining GPS error states; continuing with SV 4 as the example, elements 122, 123, 124, and 125 in the H row are set to unity. Element 99 in the H row is unity for all four SVs because the user clock affects all GPS measurements by the same magnitude.

5.8 SV Measurement Set Selection

Although only four SVs are used for measurements at any given time, positions for all 24 SVs in the constellation are continuously updated in the truth model for simulation purposes. This is necessary in order to emulate the function of an operational GPS

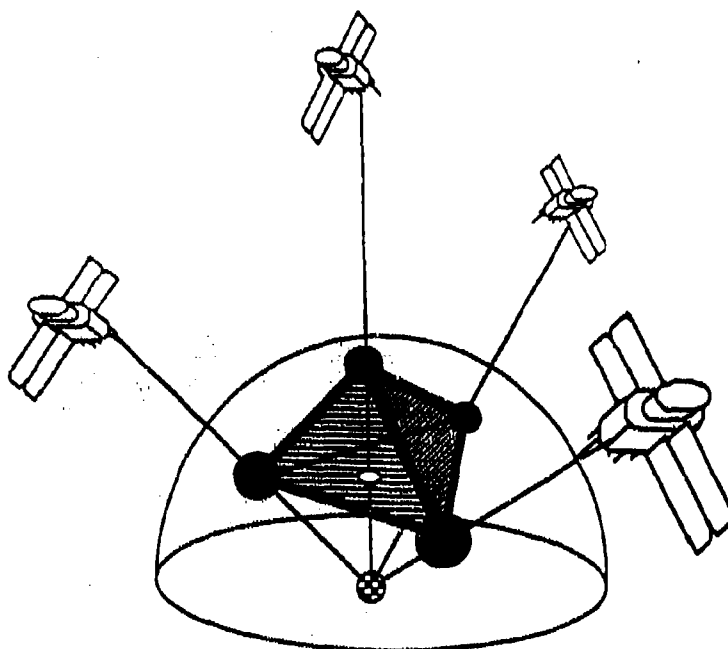


Figure 5.6. Optimal SV Set Geometry [4]

receiver. As previously noted, the GPS receiver selects the set of four SVs from *all possible combinations* of SVs in view to provide the best possible geometry for enhancing the navigation solution.

When SV and user positions are expressed in a common reference frame, the "correct" group (or set) of four SVs must be chosen from all possible combinations of SVs in view. The approach is to solve an iterative algorithm [27] which determines the optimal SV set based on geometric considerations. *Ideally*, the GPS receiver selects a set of SVs arranged such that one SV is directly overhead with respect to the user; the remaining three SVs are as low on the horizon as possible (while still permitting clear reception of their signals by the user), and they are spaced 120° apart in azimuth as seen by the user.

However, ideal geometry is seldom (if ever) achieved. Therefore, the goal is to determine which set of four SVs offers the *best* geometric configuration available at measurement times. The algorithm used solves the problem by maximizing the volume of a tetrahedron defined by the intersection of unit vectors (from user to SV) with a hemispherical surface above the user. The concept is illustrated in Figure 5.6. While it is relatively simple for a

GPS receiver to determine whether or not a particular SV is in view (either the SV signal is present, or it is not), life is not quite so simple in the simulation environment.

Because signals from SVs must be simulated, it is necessary to determine which SVs are "in view" (meaning that their signal is not "masked" from the user by physical obstructions such as the earth). Therefore, once SV and user positions are determined in a common frame at any given time of interest (say a measurement time t_i), then the elevation of each SV (with respect to the user) must also be determined. The method is to compute the elevation angle based on the unit line of sight vector from the user to each of the SVs in the optimal constellation at each measurement time. If the computed elevation angle is greater than a predetermined mask angle (5° for this research), then the SV is "in view" to the user.

The optimal SV constellation is arranged such that, depending on user position and time, a minimum of five and a maximum of eleven SVs may be in view at any given time. Further consider that any four SVs can form a measurement set. Then the number of combinations from which the best set is selected at each measurement time may range from 5 to as many as 330. Obviously, in the latter case, severe computational loading can result and "intelligent" algorithms for set selection are well worth investigation (particularly for simulations conducted on serial processors).

5.9 SV Initial Orbital Parameters

Initial conditions (orbital parameters) for the optimal SV constellation are extracted from [12] and are shown in Table 5.1. There are six orbital planes in the constellation; each plane contains four SVs. As previously noted, orbital semi-major axes are 26,609 kilometers for all SVs and the orbital period is 11 hours, 59 minutes, and 57 seconds. An added benefit to this increased orbital radius is that SVs are expected to maintain their nominal trajectories more successfully, resulting in a lower requirement for station-keeping maneuvers and a higher percentage of "in commission" time.

Table 5.1. GPS Optimal Constellation Initial Conditions [12]

Satellite ID	LAN	Alpha	Satellite ID	LAN	Alpha
01	325.730284	190.96	13	145.730284	312.30
02	325.730284	220.48	14	145.730284	340.93
03	325.730284	330.17	15	145.730284	87.06
04	325.730284	83.58	16	145.730284	209.81
05	25.730284	249.90	17	205.730284	11.90
06	25.730284	352.12	18	205.730284	110.76
07	25.730284	25.25	19	205.730284	143.88
08	25.730284	124.10	20	205.730284	246.11
09	85.730284	286.20	21	265.730284	52.42
10	85.730284	48.94	22	265.730284	165.83
11	85.730284	155.08	23	265.730284	275.52
12	85.730284	183.71	24	265.730284	305.04

5.10 Summary

This chapter presents the basic concepts related to the GPS subsystem and introduces the GPS dynamics error model. In addition, the GPS pseudo-range measurement model equations are developed in detail. The conventional *difference* measurement approach is used once again. Finally, SV orbital calculations are presented along with initial conditions, and the criteria for set selection. The approach is a considerable departure from previous research [11, 19, 32, 33] due to the addition of a full GPS constellation in which orbits are modeled explicitly, and because of the addition of crucial functions (PDOP calculations, set selection, and set switching algorithms) which emulate the operation of a GPS receiver.

VI. Results

Prior to a moderately detailed analysis of the results obtained for the considerable number configurations tested, several items of general interest which affect most or all of the simulations are discussed. In addition, some of the major differences between this and previous research are highlighted.

6.1 Monte Carlo Analysis of the NRS Error Model

The research conducted in this thesis hinges on the development and *employment* of a system (*truth model*) which is mainly constructed from models for the LN-93 INS, the RRS, and the GPS subsystems. The fact that this truth model is constructed in software (embedded in MSOFE [5]) represents a significant extension to previous research [31, 33] in which the truth models were not installed in the MSOFE "system" to create an environment in which various Kalman filter designs could be tested. (Rather, previous truth models were tested in the MSOFE covariance mode. Previous researchers used actual data obtained from empirical sources related to CIRIS operation to drive the filter designs contained in [31, 33].) The truth model developed in this research, along with trajectory data generated by PROFGEN [1] and the new SV orbit calculation software, *generates* measurement data as well as reference variables which are used to test the performance of the (full-order) Kalman filter, and constitutes an *environment* in which a variety of full-order and reduced-order Kalman filters may be tested against a common, high-fidelity standard.

Several system-level configurations (presented below) are tested and analyzed in this research. Generally, analysis of the configuration of interest consists of performing a series of 10 alignment runs, followed by a series of 10 flight simulation runs, and observing the stochastic time history of error-state variables of interest. (In the case of very large dimension models with many measurement updates, fewer runs may be used, resulting in lower confidence in the data sample statistics thus derived [22]. Specific instances are clearly identified where they exist.)

At the beginning of each alignment run, the *truth model* state vector is initialized in Monte Carlo fashion: each truth state is set to a random value based on the state's initial covariance (and a random number from a pseudo-random number generator). At the end of the alignment runs, terminal conditions for the truth and filter state vectors and the final covariances are written to a data file. This data file is then used to provide initial conditions for flight runs. Thus, a true Monte Carlo fashion simulation is preserved for the entire analysis sequence.

6.2 Feedback and Platform Alignment

As noted above, the *truth model* state vector elements are initially randomized in a manner intended to represent actual error conditions which may be present in INS, RRS, and GPS subsystems. The intention is to determine the effect of the randomization of truth model states on the Kalman filter's ability to perform its estimation task. This (stochastic) stress test for the Kalman filter is another major difference in this and previous research.

As a result of this stochastic stress test, an interesting (but crucial) discovery was made concerning the truth model. Recall that the INS model consists of error states. For instance, state number four is the error in north platform tilt. Having initialized the truth model error states in the Monte Carlo fashion described above, the system is allowed to propagate its states for the period of the eight-minute alignment. During the alignment, the Kalman filter is provided with measurements to improve the estimation process. However, by the end of the alignment period, the *truth model* has typically developed state variables of an inappropriately large magnitude. For example, the latitude and longitude error states grow to the order of 20,000 feet during an eight-minute alignment.

The natural question to raise here is whether or not such behavior should be expected. In reply, this sort of behavior might easily be observed in a physical INS that is turned on but NOT torqued to local level. Typically, errors of this magnitude are undesirable after having completed an *alignment*. Even though the Kalman filter can, with *some* forms of measurement updating, maintain a reasonably good estimate of the misalignment error states, the INS would subsequently begin a flight run (navigation mode) with large errors

rather than with minimal errors that are seen in physical systems that are "leveled" during the alignment process.

One might be inclined to consider that the error magnitude is immaterial because the interest is simply to maintain a good estimate no matter what the error magnitude happens to be. Two counterpoints can be made to that argument. First, serious numerical complications invariably arise when attempting to propagate tightly coupled error models in which state values are becoming increasingly large. In such a case, numerical precision is crucial but often impossible because computer wordlength becomes a serious limiting factor [22]. In addition, the linear perturbation model adequacy is placed in serious jeopardy as error magnitudes become excessively large [26].

How can the issue be resolved? Two approaches are used successfully in this research. First, one might choose to use feedback during the alignment process [22]. In this approach, the Kalman filter state estimates are fed back to the truth model. This technique has the effect of "leveling" the platform by reducing the magnitudes of the truth model error states. The approach does not appear to work as well using full state feedback as it does when using only partial state feedback (i.e., only position, velocity, and tilt states are fed back). This behavior may stem from the fact that there exists a problem with observability in the INS error model. For instance, velocity measurements affect only the "basic" nine error states (position, velocity, and tilts), and provide *no new information* to improve the Kalman filter estimates of the remaining *eighty-four states*. Additionally, one must consider that feedback of some states is impractical due to the *inability* to correct errors which may be estimated correctly but which are physically inaccessible. As an example, the atmospheric error associated with RRS transponders can be estimated very well, but one cannot change the atmospheric properties which create the error; feedback of states in this category is not only impractical, it can actually exacerbate estimation errors.

The second approach is to use an "impulsive reset" [22] of the truth model error states. The first requirement is for the Kalman filter to acquire high quality estimates of error variables. Subsequently, instantaneous corrections are applied to the truth model states (meaning that data registers are reset, based on the available feedback states). It is similar to the feedback approach discussed above, and differs primarily in the frequency

at which feedback is applied.

The "impulsive reset" method is adopted for general use in this study. Investigations reveal that the two methods provide comparable performance in reducing the truth model error magnitudes, but also (not surprisingly) revealed a noticeable improvement in software execution time when the second approach is taken. Consequently, a *single* impulsive reset is applied at the end of each alignment run in order to initialize flight runs with an appropriately "level" platform. Once again, only the basic nine error states are fed back.

6.3 Performance Analysis Baseline

A complete error-state model for the Navigation Reference System (NRS) is developed in Chapters III through V. The foundation is laid with the 93-state LN-93 RLG-based INS model. The basic INS model contains a single baro-altimeter state which is subsequently revised to improve model fidelity. Next, the RRS transponder model is added to the INS model. The NRS model is completed with the addition of a GPS model that incorporates a full constellation of 24 moving space vehicles.

The results achieved from simulation and analyses at each of the junctures described above are now presented. Key variables (such as position, velocity, and platform tilt errors) are plotted for each of the test configurations. The plots are contained in Appendices C through I. Salient features are discussed in the sections which follow.

It is important to begin research of this type with a "baseline" to which subsequent performance changes (resulting from changes to the model) are compared. Consequently, the first configuration which is tested and analyzed is that of the basic LN-93 INS. The Litton error model contains 93 states, including a single baro-altimeter state.

The complete 93-state error model is programmed into the MSOFE truth model using initial conditions suggested by Litton [20], and a full-order Kalman filter is constructed based on the same model. The model is then tested in two distinct simulations. First, an "alignment" series (10 Monte Carlo runs) is performed. Initial INS position is assumed to be 45° North latitude and 0° longitude. [These conditions are chosen for consistency with Litton conditions.]

The second test series is a 10-run Monte Carlo simulation of the expected INS performance under flight conditions. Variables from the fighter flight profile presented in Chapter III are used as the nominal conditions about which the truth model equations are relinearized during the propagation cycle. The extended Kalman filter is relinearized about its best estimates of the trajectory variables.

This phase of the model analysis is conducted to ensure that the baseline model performs in a manner which is consistent with the performance stipulated by Litton for the LN-93. During the alignment series, both velocity and baro altitude measurements are provided to the Kalman filter. For the flight series, only baro altitude measurements are used.

In referring to Appendix C.1, the reader will find several plots depicting the error behavior of the 93-state INS during alignment. As noted previously, the LN-93 error states are "randomized" at the start of each alignment - thus 10 different random initial state vectors are used to seed the 10 Monte Carlo alignment runs. The center plot trace (- - -) represents the mean error time history for the indicated state. Mathematical descriptions for data time histories are included at the beginning of Appendix C.

The true standard deviations of the indicated mean error variable is represented by σ_{true} . The traces (. . .) which bound and "track" the mean error time history represent the mean error plus and minus σ_{true} . The final pair of traces (—) represents the filter-computed $\pm \sigma_{filter}$ for the error variables indicated. They are symmetrical about zero because the Kalman filter "assumes" that its errors are zero-mean [22]. The filter-computed error standard deviation magnitudes (σ_{filter}) may be compared to similar plots in the Litton reference [20]. In the case of the position states (latitude, longitude, and altitude) the comparison is excellent; the magnitudes of the σ_{filter} plots for this research compare very closely to the Litton results [20].

The only significant difference is in the vertical velocity error state. The magnitude of its σ is somewhat lower here than in the Litton document [20]. This occurs due to the fact that Litton uses only horizontal velocity updates during alignment. In this research, vertical velocity and baro-altitude measurements are used in conjunction with horizontal

velocity measurements.

Initially, vertical velocity measurements are adopted because of very large magnitude errors (discussed previously) that developed during alignments. It is postulated that the random initial condition effects could be reduced or eliminated by providing additional measurement information. Although the referenced errors are not adequately quelled by these additional measurements, a significant improvement in the vertical channel states was noted (i.e., vertical velocity σ decreased by about 50 percent compared to the Litton results). Consequently, vertical velocity and baro-altitude measurements are used in alignment runs throughout this research.

Two noteworthy features appear in the 93-state alignment plots contained in Appendix C.1. First, the latitude and longitude error time histories tend to drift away from the expected zero-mean. This result occurs because of the lack of *horizontal* position information during these alignments (notice that the mean altitude error looks "reasonably" zero-mean - a consequence of incorporating baro-altimeter measurements which is another departure from the Litton alignment procedure).

The second feature which is likely to catch the reader's attention is the flat-line (zero) response of ΔS_1 . This is the characteristic behavior of this vertical channel aiding state during INS alignment. This response occurs because ΔS_1 is a function of altitude rate (which is nominally zero during alignment). (The behavior of ΔS_1 is far more interesting in the flight runs.)

In Appendix C.2, the reader will find the mean error and standard deviation time histories for several LN-93 variables which result during a fighter flight simulation. Once again, a 10-run Monte Carlo series is run for the flight simulation. Initial conditions for the flight runs are those *final* conditions which resulted from the alignment runs described above.

The reader may note that the horizontal and platform tilt error states (still) do not look zero-mean. This is again due to the lack of horizontal position information. [For this series of runs, baro-altitude measurements are used to bound vertical channel errors. No other measurements are available to improve the Kalman filter state estimates.]

In addition to the large magnitude mean errors, the filter-computed error standard deviations for many states (vertical channel excepted) are roughly twice as large as the same quantities in the Litton document. Note that this statement does not imply that the truth model behavior is aberrant; it means simply that the 93-state Kalman filter *based* on the truth model is not performing well as a state estimator. This is not unexpected; without adequate measurement information, and considering the previously noted observability issues inherent to this problem, a Kalman filter cannot be expected to perform well under such highly dynamic conditions as those simulated in the fighter flight profile. However, performance can be improved as indicated in subsequent simulations.

Although not contained in the appendix, the 93-state *truth model* covariances were plotted and compared to Litton results [20]. The excellent agreement between those results and Litton predictions constitutes additional validation that the truth model performs in a manner which is consistent with expected INS performance. The reader should note that this performance is only achieved by using feedback *during* the alignment - results are much different without feedback.

As promised, the behavior of state 13, ΔS_1 , is quite interesting in the flight runs. The mean error, true standard deviation, and filter-computed standard deviation all have the appearance of switching on and off. This is in fact the case. Recall that ΔS_1 is a strong function of altitude rate. It is "on" during intervals of altitude change, and "off" during flight segments at constant altitude. The state values, and consequently the standard deviation, become zero.

6.4 96-State Error Model Performance

After adding the additional baro-altitude states discussed in Chapter III, the same sequence of testing (alignments followed by flight runs) is applied. The goal in this case is merely to assure that, by enhancing fidelity through the addition of states in the truth model and Kalman filter, no performance degradation is induced in the vertical channel states. This step is viewed as a "quality control check" prior to making subsequent additions of subsystem module (i.e. RRS and GPS). Comparing the alignment plots obtained here to those in Appendix C.1 (93-state alignment), the reader is apt to conclude

(Incorrectly) that the plots are identical. Closer inspection reveals the differences. Note particularly the azimuth error state (Figures C.3 and D.3) and the scale differences on the vertical velocity state (Figures C.4 and D.4), and the altitude states (Figures C.5 and D.4).

The slight differences in the vertical states are a direct reflection of the change in the baro-model, while the azimuth error anomaly is simply due to a difference in the initial condition for the state which had no effect on steady state operation during alignment runs. The additional baro states have no apparent effect on horizontal position (latitude, longitude) errors during the alignment runs.

However, the differences are slightly more obvious in plots from *flight* runs for the 96-state model. In this case the filter-computed standard deviations match Litton predictions quite closely. This is primarily due to using slightly increased noise magnitudes for the Kalman filter (as opposed to the noises strengths set forth in the Litton document which are *still* used in the truth model). In this research, it is determined that an *experimentally tuned* Kalman filter appears to achieve better altitude channel performance when a factor of 1.2 to 1.5 is applied to the nominal noise strength suggested by Litton.

An obvious difference is apparent in the vertical channel states (compare Figure C.11 to D.11). The large changes in the true baro-altimeter error, depicted in Figure D.11(b), drive similar changes in the INS altitude error, shown in Figure D.11(a).

The baro-altimeter changes stem directly from the addition of the new altimeter error states. The most significant source of the true error is the scale factor error state which is included in the truth model but is not included in the filter. Early investigations revealed that the addition of this state to the filter model resulted in severe numerical difficulties in the filter covariance propagation carried out by MSOFE.

In many instances, particularly during rapid altitude transitions, either integration errors or negative variances (albiet extremely small magnitude) resulted. This distressing result (negative variances are mathematical anomalies [22]) is directly traceable to the vertical channel aiding state, ΔS_1 , which varies strongly with altitude and altitude rate.

The result is that ΔS_1 behaves quite erratically during rapid altitude change or at high altitude.

Three options are open to resolve the problem. First, greater numerical precision [i.e. double precision variables] would likely reduce or eliminate the tendency to obtain negative variances [25]. Unfortunately, increasing numerical precision also results in significantly increased computational burden [22]. Second, software may be written to detect negative covariances and (cautiously) reset them to zero. This approach is also undesirable because it introduces the *possibility* of masking other problems which may be present in the Kalman filter [25]. Therefore, the third option is taken. The problematic scale factor state is removed from the Kalman filter while leaving the truth model unchanged.

The result is plainly evident. Baro-altimeter error in the truth model depends on scale factor error while the Kalman filter omits this error source. Consequently, during high altitude portions of the flight runs, large estimation errors occur in the baro-altimeter channel, with attendant estimation error in the INS altitude channel. Fortunately, the estimation error is not unacceptably "bad" and does not seriously degrade the overall performance of the Kalman filter, as evidenced by the consistency of the horizontal position and velocity states.

With a revised baro-altimeter model now integrated into the truth model (although not in the Kalman filter), the next step is to add RRS transponder aiding. In preparation for that step, the 96-state model is re-run. This time, however, different initial coordinates are used for alignment and flight run initializations. Previously, initial latitude, longitude, and altitude were set to 45° north, 0°, and 0 feet respectively. For this set of runs, the initial conditions are those of Holloman AFB. (Latitude is north 32.78°, longitude is 105.99° West, and altitude is 4100 feet.) This change is necessary due to the fact that the actual locations of the RRS transponders used in subsequent configurations are on or near Holloman AFB.

The differences between mean error time histories from this (refer to Appendix E) and the previous case (Appendix D) are generally unremarkable. Performance is very consistent with that achieved previously. Once again the filter-computed mean error standard deviations are also consistent with Litton predictions [20]. Just as in the previous case,

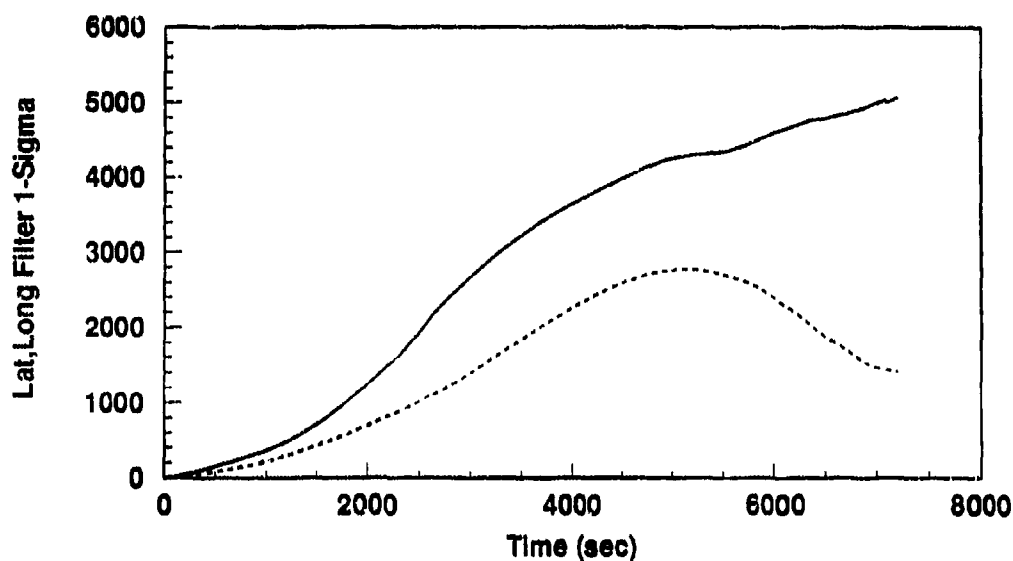


Figure 6.1. 96-State Model, 2-Hour Flight: Latitude (—), and Longitude (---) Filter-Computed σ Plots

latitude and longitude mean errors are not "white", due to lack of horizontal position information.

In order to facilitate a direct comparison of the horizontal positions states between this and subsequent configurations, Figure 6.1 is included. In Figure 6.1, latitude error is indicated by the solid trace and longitude error is depicted by the dashed trace. The behavior exhibited in these plots is quite comparable to latitude and longitude error plots in the Litton LN-93 documentation [20].

6.5 72-State Error-Model

Occasionally, a step forward must be preceded by a step (or two) backward. Initially, the RRS transponder model (see Chapter IV) was successfully added to the 96-state INS model to form a 122-state truth model. Following that step, the GPS model (Chapter V) was added to the INS/RRS model to form a 152 state truth model. A quick discovery was made at that point. The accelerator board (referenced in Chapter I) cannot handle a problem of this size. Recall that with 152 states, the $U - D$ covariance matrix has $152(153)/2 = 11,628$ nonredundant terms which must be continuously calculated.

Whether because of data storage needs or executable code size requirements for random access memory (RAM), the accelerator board is unable to compile and link the software to create executable code.

This problem is peculiar to the board. This point is verified by successfully compiling and running under a different operating system. However, in the interest of preserving the speed advantage gained by using the accelerator board, a decision is made to reduce the truth model and Kalman filter dimensions sufficiently to circumvent the hardware limitations which are encountered.

References to previous work [11, 19, 31, 33] suggested that the INS states (originally numbered 70 to 93 by Litton) are "expendable." A series of Monte Carlo alignment runs are made to verify that these states could legitimately be eliminated without significantly affecting the error behavior of the truth model.

The plots contained in Appendix F directly compare the performance of the reduced truth model with the full 96-state model. In all cases the reduced-order truth model appears to have "captured the essence" of the full-order model behavior. Minor variations are evident in the altitude and vertical velocity states. However, the difference is mainly an artifact of the scale. Virtually all of the difference may be attributed to the simple fact that these results come from 10-run simulations. If more runs are used in the Monte Carlo simulations (possibly 25 to 50 [5, 22] runs), the differences may diminish significantly. With just 10 runs, the agreement between the two candidate truth models is generally excellent.

As an aside, the 96-state truth model covariances which are plotted in Appendix F may be compared to Litton results [20]. The excellent agreement between these results and Litton predictions constitutes additional validation that the truth model is performing in a manner which is consistent with expected INS performance. The reader should note that this performance is only achieved by using feedback *during* the alignment - results are much different without feedback.

The obvious extrapolation (given the excellent agreement between the 96-state and 72-state models) is that the 72-state model is performing its INS emulation function with

adequate fidelity. Therefore, the 72-state model is adopted to represent the INS error-state subsystem which is integrated into the overall NRS truth model. The additional states (above the 69 Litton states which are retained) are the new baro-altimeter states.

6.6 98-State Error-Model

Combining the 26-state RRS model with the 72-state INS model produces the 98-state model which is tested to determine the baseline INS/RRS performance. Appendix G contains the mean error, mean error $\pm \sigma_{true}$, and $0 \pm \sigma_{filter}$ time history plots similar to those shown in previous configurations. In addition to the variables that are plotted in previous configurations, results that are representative of RRS transponder error states are also plotted in Appendix G.

The impact of providing RRS measurements to the Kalman filter is dramatic indeed. An obvious improvement for the alignment simulations is evident in the horizontal position channel estimates (refer to Figure G.9). Latitude and longitude mean errors are now much closer to the zero-mean predicted by Kalman filter theory. In other cases, where the mean error is not zero-mean (i.e., the azimuth error state, Figure G.3), the result is very likely to be due to the fact that these data are obtained from 10-run Monte Carlo simulations. For improved statistical validity, 25 to 50 or more Monte Carlo runs are generally considered necessary [5, 22]. Careful comparison of the vertical scales for horizontal channel states shows that the addition of RRS transponder measurements during alignment reduces the true- and filter-computed one sigma values by more than 50 percent. Additionally, there are no significant biases or ramps evident in the plots.

A far more substantial effect is evident in the flight runs. With RRS transponder measurements, horizontal position errors are reduced from several thousand feet (in previous configurations) to less than 40 feet during all phases of the flight. The significant variations in the true and filter one sigma values are due to two sources. First, significant transitions occur in the altitude, velocity, and tilt states during periods of high dynamic maneuvering. Second, the aircraft range from the transponders has an obvious effect in horizontal position errors. Flight regimes in which the aircraft is at low altitude or is a long distance from the transponders result in increasing positional uncertainty. When the air-

craft is high overhead with respect to the transponder positions, much better estimations are possible.

Because problems persist in the vertical channel, and with the publication of a paper by Litton engineers on the topic [2], the scale factor state is (in this and subsequent configurations) essentially eliminated from both the truth and filter models. Although obviously an undesirable approach, the scale factor state was simply initialized to zero for all runs. This change accounts for the improved appearance of the altitude error estimation in the later configurations (note the improvement in the altitude channels in Appendices G.2, H.2, and I.2 as compared to Appendices D.2 and E.2). This step is taken in the interest of continuing research with minimal interference from the newly installed and possibly invalid [2] baro model.

In all cases, the filter appears to be reasonably well "tuned" in that the filter one sigma values bound the mean error $\pm \sigma_{true}$ traces the majority of the time, without being excessively conservative. [Conservative, as used here, denotes the condition in which the magnitude of the filter computed mean error σ is larger than the true σ magnitude.]

In order to facilitate a direct comparison of the horizontal positions states between this, previous, and subsequent configurations, Figure 6.2 is included. In Figure 6.2, latitude error is indicated by the solid trace and longitude error is depicted by the dashed trace. Note the significantly reduced σ magnitudes in this configuration compared to Figure 6.1. Performance is improved by roughly three orders of magnitude when compared to the INS performance when aided by baro-altitude only.

6.7 128-State INS/RRS/GPS (NRS) Model

In the final phase of integration, the GPS model is added to the 98-state INS/RRS model, bringing the total number of NRS states to 128. [This number was previously determined to be below the threshold at which the accelerator board stops cooperating.]

Two modes of testing are performed for the GPS addition. First the alignment and flight runs are performed with GPS measurements; no RRS measurements are included. Second, the alignment and flight series are re-run using both GPS and RRS measurements.

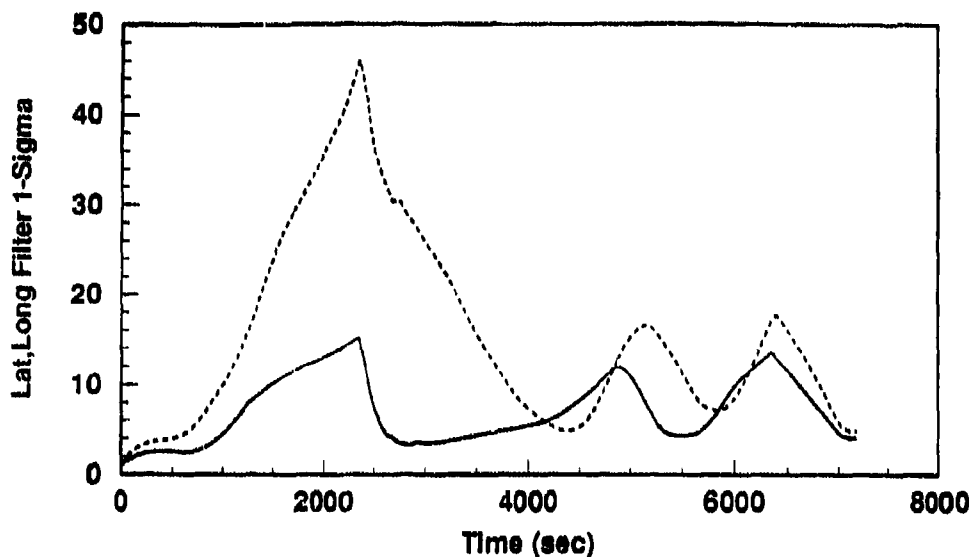


Figure 6.2. 98-State Model, 2-Hour Flight: Latitude (—), and Longitude (···) Filter-Computed σ Plots (Note: RRS Measurements Only)

6.7.1 *128-State NRS Model: GPS Measurements Only.* Results from the 10-run alignment simulation of NRS aided by GP5 (and the usual velocity and baro) measurements are shown in Appendix H.1. Performance is slightly degraded compared to the 98-state RRS-aided alignment (Appendix G.1). Even though the σ_{true} values are not substantially different from the previous case, the filter is estimating its own errors less effectively. The difficulty seems to be limited to the horizontal position states. Other states appear to be equally well-tuned as in the previous case. The first reaction may be to adjust the driving noises associated with the latitude and longitude states. This approach may work, but extreme caution is warranted. Another postulation is that single precision numerical operations are inadequate for a problem of this type. Such may well be the case. However, double precision calculations require a significant increase in software execution time. This point is addressed in more detail in Chapter VII.

Two factors must be considered. First, the nine basic error states include *error angle* states in the LN-93 model. The correlation between error angle states and navigation frame position errors does not facilitate the tuning process. (Recall the position error transformation presented in Chapter IV to obtain $[\delta\lambda, \delta L, \delta h]$ from the error angle vector $[\delta\theta_x, \delta\theta_y, \delta\theta_z]$.) The hazard is that "tuning" one of the error angle states generally affects

all of the position states - sometimes in an undesirable fashion. The extent of the hazard goes beyond that, however. Because of the highly coupled nature of the basic nine error states, adjusting the noise level on one state may cause adverse effects on several others. Caution and patience are the only solutions to the tuning problem.

Three one-hour flight runs are performed for the 128-state NRS model (compared to 10 two-hour runs in previous cases). The execution time for a problem of this size is a major concern. As an example, the 10-run Monte Carlo simulation for the alignment runs takes approximately 15 to 18 hours. Consider that the alignments (480 seconds each) are about *one-fifteenth* as long as the flight runs. Additionally, flight trajectory data interpolation increases flight simulation times significantly. As a result, a 10-run flight simulation series is estimated to require a minimum of 225 hours (more than nine days) to complete on the fastest processor generally available at AFIT. [There is hope for the future. This issue is addressed again in Chapter VII.]

The reduction in flight time is really not a serious drawback. In earlier cases, the two-hour flight profile is used simply to facilitate direct comparisons of the configuration under test and the Litton reference [20]. Such comparisons may just as easily be made for one-hour flight times. The validity of this point is evident if the reader observes the error behavior exhibited in the RRS aided INS case (Figure G.9). The maximum estimation errors occur at approximately 2600 seconds (during rapid, high-g maneuvering far from the transponder sites), and have returned to lower magnitudes by the end of the first hour of flight.

However, the reduction to three runs in the Monte Carlo series is cause for concern [22]. In cases such as this, confidence in the validity of the statistical data is reduced. (In fact, 10-run series for problems of this type are considered to be marginally adequate in a statistical sense [22].) Therefore, quantitative comparisons of performance based on three runs should probably be avoided. On the other hand, *cautious* qualitative comparisons are probably still reasonable.

With that qualification, a comparison between this and the previous configuration (pre-GPS) is presented. The 3-run flight results in which GPS (but no RRS) measurements

are used to aid the INS are shown in Appendix H.2.

The general characteristic which seems to be evident in virtually all error states (at least those whose time histories are plotted) is that the filter-computed σ magnitudes appear to be consistently lower for the GPS-aided INS, compared to the RRS-aided INS. However, the true σ values are somewhat larger than previously and the mean error time histories are not satisfactorily bounded by the filter-computed σ magnitudes. This result is attributed to two factors. First, data plots are based on 3-run Monte Carlo flight simulations. Second, numerical problems which sometimes result from using single precision versus double precision are quite possibly at fault. (This issue is also addressed further in Chapter VII.)

In order to facilitate a direct comparison of the horizontal positions states between this, previous, and subsequent configurations, Figure 6.3 is included. In Figure 6.3, latitude error is indicated by the solid trace and longitude error is depicted by the dashed trace. Note the significantly reduced σ magnitudes in this configuration compared to Figure 6.2. In some flight regimes (notably those far from the transponder sites, or those with depressed elevation angles between the user and transponders) performance appears to have improved by about one order of magnitude when compared to the INS performance achieved with RRS aiding alone.

6.8 128-State INS/RRS/GPS (NRS) Model

6.8.1 128-State NRS Model: GPS and RRS Measurements For this configuration, both RRS and GPS measurements are used. For the alignment simulations, 10 Monte Carlo runs are performed. However, only a single Monte Carlo run is performed for the flight simulation. (Therefore, extreme caution must be exercised in any analytical conclusions drawn from the flight results.)

Alignment results look quite similar to results obtained with either GPS or RRS alone. The general trend is a very slight reduction in the filter estimates of the overall error magnitudes during the alignment phase; no remarkable behavior is observed.

On the other hand, flight results are not quite as optimistic. Although it is clear that

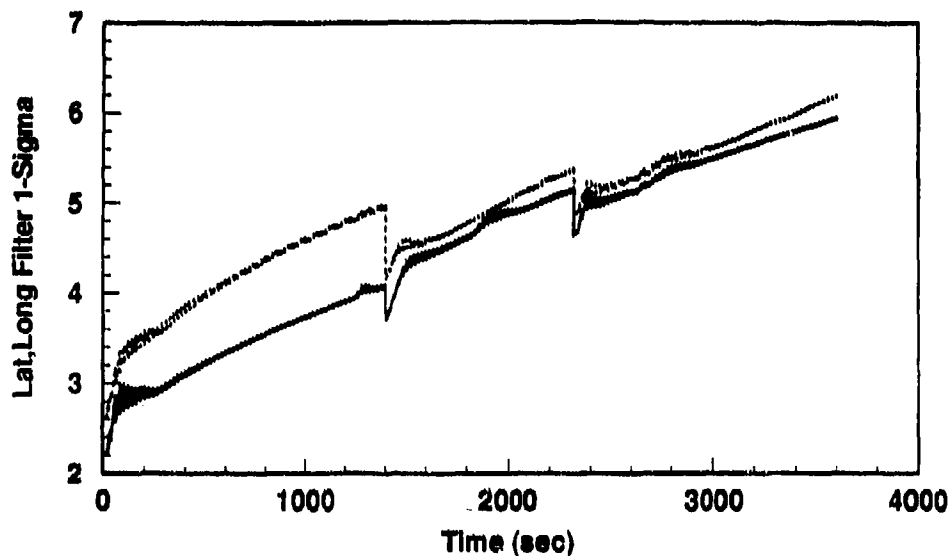


Figure 6.3. 128-State Model, 2-Hour Flight: Latitude (—), and Longitude (···) Filter-Computed σ Plots (Note: GPS Measurements Only)

the filter “thinks” that it is doing a better job of estimating the variables of interest, it is in fact not doing as well as it thinks. The filter, aided by both RRS and GPS, “thinks” that the estimation errors are smaller than they are. Its estimates of error variances are consistent with *expectations*, but are inappropriate for the situation. For all error states, the mean error time history trace should be bounded (68 percent of the time) by the filter-computed σ . This is not the case for some states, notably those of horizontal position errors.

The apparent difficulty in this case is attributed to two factors. First, one must return to the issue of using single-precision versus double-precision for a problem of this nature. Such a choice may easily account for some of the unexpected filter behavior. Additionally, the fact that a single flight run is presented is cause for carefully considering the data which are thus obtained. Many of the “temporary divergences” which occur at various high dynamic points of the flight regime are statistically “smoothed” when a larger number of runs are used.

In order to facilitate a direct comparison of the Kalman filter horizontal positions states between this, previous, and subsequent configurations, Figure 6.4 is included. In

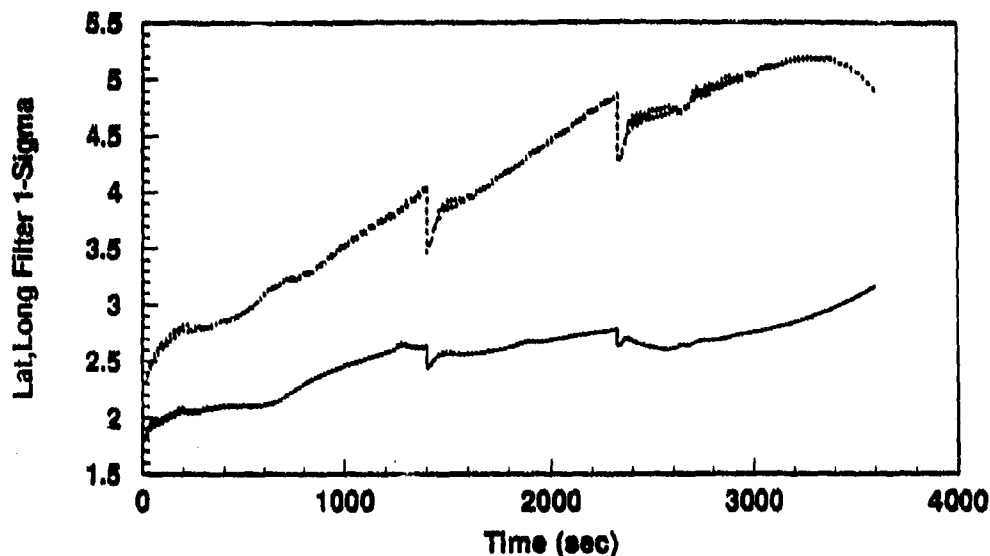


Figure 6.4. 128-State Model, 2-Hour Flight: Latitude (—), and Longitude (---) Filter-Computed σ Plots (Note: RRS and GPS Measurements)

Figure 6.4, latitude error is indicated by the solid trace and longitude error is depicted by the dashed trace. The σ magnitudes in this configuration are slightly reduced compared to Figures 6.2 and 6.3. The slight ramping effect is exacerbated by the plot scale. Although such behavior is not consistent with theory [17], it may be explained by the small magnitude pseudo-noises which are used in the filter model for the GPS position states. The noise strength may be somewhat higher than necessary to preclude the filter covariances from decreasing to zero. Experimentation with these "tuning" parameters is warranted.

After reviewing the configurations involving GPS measurements, two conclusions are drawn. First, GPS aiding alone appears to offer a noticeable improvement over simple transponder-aided INS. In several flight regimes, the GPS-aided solution is significantly better than that of the RRS-aided INS. This fact is a direct result of two features which are inherent to this simulation (and which may not necessarily be true in general). (The flight path extends far beyond the optimal coverage areas for the fixed transponders, and only six transponders are used; many actual flights use twenty or more transponders.) Second, the true Kalman filter estimation errors are not adequately bounded by the filter-computed σ (as they are in previous cases) for the simulations in which GPS measurements are used.

The apparent problem with the true estimation errors may be attributed to numerical precision problems resulting from single-precision computations. The desire to avoid double-precision simulations is previously discussed. However, it is possible that this reticence resulted in less than spectacular filter performance. As in most engineering situations, the "trade-offs" must be carefully considered and the potential for problems understood.

Another important conclusion which is tentatively drawn is that the configuration in which GPS and RRS measurements are used together offers the best overall performance. Although the true error behavior does not appear to be consistent with filter predictions, simple filter tuning or an increase in numerical precision may be sufficient to reduce or eliminate the disparity.

6.9 Summary

This chapter presents the results from a large number of configuration-dependent Monte Carlo simulations. Each configuration is tested in both an alignment and flight simulation. General results from each case are discussed, with reference to particular plots (contained in appendices and distributed in the text) for emphasis of important points. The reader is cautioned of the potential for observability and numerical problems which may stem from simulations of the type conducted here.

VII. Conclusions and Recommendations

This effort is concluded with some thoughts on its objectives, the degree of success achieved, and the potential to spawn future research. The major goals of this research are closely tied to the ultimate objective of providing the Central Inertial Guidance Test Facility, Holloman AFB, NM, with a superior navigation reference system that can be used as the standard to which all other navigation systems are compared. This is a non-trivial objective which demands the "optimal" integration of several highly complex systems and which encompasses a large body of knowledge from several engineering disciplines. The variety of disciplines and technologies needed to construct such a reference system successfully is at once exhilarating and staggering.

7.1 Baro-Altitude Model

In the attempt to further the broader goal stated above, the goal of improving the baro-altitude portion of the LN-93 error model is undertaken. Though innocuous at a glance, this portion of the project turns out to be far more troublesome than anticipated, confounds a great deal of effort and consumes an inordinate amount of time which would have been better devoted to other pursuits.

It appears that this research validates a result that Litton has recently published in a paper presented at the National Technical Meeting of the Institute of Navigation, January 1991 [2]. In essence, the paper states that the vertical channel model included in the original LN-93 document was not fully evolved, and a revised model is presented.

In addition to pointing out how the new model differs from the previous Litton model, the paper also describes a scale factor model that is totally incompatible with the scale factor state developed in the revised baro model presented in Chapter III. Because the vertical channel issue had already been "put to rest" in this research when this fact came to light, the model was not redesigned. Rather, the scale factor state was simply initialized to zero for all runs. This change accounts for the improved appearance of the altitude error estimation in the later configurations (note the improvement in the altitude channels in Appendices G.2, H.2, and I.2 as compared to Appendices D.2 and E.2). After

"zeroing" the scale factor state, the truth and filter models are again [effectively] defined identically; neither includes the scale factor error model. The choice to set the scale factor contribution to zero is not correct; it is merely expedient to preclude interference with the other major objectives of this work. It is fully recognized that a more appropriate model is required.

Consequently, revision of the baro model is both a success and a failure in terms of thesis objectives. The baro model was indeed revised and was implemented in the truth model and, to a lesser extent, in the filter model. Its operational characteristics including the tendency toward numerical problems in the Kalman filter were identified. However, new facts have now surfaced which indicate that the proposed model was not entirely appropriate [2]; the result is that the new model is revised to contain three states rather than four.

The solution to the difficulties with the vertical channel states is (hopefully) quite simple. Future researchers in this area must be provided access to a current error model for the LN-03, including the newly revised vertical channel mechanization.

7.2 GPS Model

Another goal established at the outset was to devise software to calculate SV orbital positions for the interval of the simulated aircraft flight run. This goal is completed. In fact, software of a *general* nature is fully developed and tested to solve the SV orbital mechanics problem for *arbitrary* intervals of time, and to find and select the best set of SVs for performing GPS measurements in a moving vehicle (user). These achievements represent significant extensions of previous research.

7.3 Truth Model

The major goal of assembling and testing a high-fidelity NRS truth model which is embedded in MSOFB is also completed. The complete error model is thoroughly documented and is fully operational in software. This provides an environment within which future Kalman filter designs may be evaluated. This achievement is another major step

beyond previous research and is particularly important because it provides an environment in which reduced-order Kalman filter designs may be tested prior to *operational* use at CIGTF.

As a first step in the future, AFIT research should focus on achieving a significant reduction in the number of states in the Kalman filter used to estimate NRS errors. This step is not only important for potential operational considerations, but will also aid tremendously by "creating space" for other enhancements to the NRS model. (One such enhancement is differential GPS.) Additionally, state reduction in the Kalman filter is expected to have the advantage of reducing the observability problems which have been noted to exist in the full-order Kalman filter. Observability analyses (see Section 7.5) will assist in state reduction efforts.

Inherent to the state reduction process is the need to establish the adequacy of the reduced INS truth model. (Recall that the INS truth model is reduced from 96 to 72 states in this research.) Although a 10-run Monte Carlo series is performed to show that the 72 state model fidelity is adequate, further analysis may be warranted. It is recommended that future researchers run the 72 state Kalman filter against the 96 state INS truth model and examine the residuals for appropriate statistical properties [26] to establish conclusively if the chosen state reduction is adequate.

7.4 Monte Carlo Analyses

To the maximum extent possible, the requirement to determine performance characteristics (via Monte Carlo analyses) for each of the previously defined subsystem and system level combinations has been satisfied. A series of 10-run Monte Carlo series is performed for *ten of twelve* configurations. In the other two cases (the largest dimensions in the truth model and Kalman filter), a three-run series is performed for qualitative analysis. It seems fair to say that the goal is largely satisfied, although it cannot be claimed that the two series which contained only three runs are conclusive proof that NRS functions as well as desired.

7.5 Opportunities for Future Research

Some topics which are, necessarily, given cursory treatment in this research might easily form the basis for extensive research in the future. One such area is that of error budgeting [22] and optimal tuning of the NRS Kalman filter. As noted previously, tuning a filter of the size represented here is a formidable task. However, it is one which should be undertaken. The major requirement is access to significantly more computational power than that which is generally available to AFIT students. Options include the new Sun Sparc-Stations (operating at 28 MIPS) which are expected to be "on-line" in the near future. In addition, every effort should be made to use double-precision variables and to perform all calculations in double-precision as well. Many of the numerical problems which are evident in this research may be eliminated by using double-precision variables.

Another suggestion that goes hand-in-hand with error-budgeting is that of Kalman filter state reduction. As noted in Section 6.5, a highly desirable performance characterization test is to run any proposed reduced-order Kalman filter against the full order truth model and observe the statistical properties of the residuals [26]. Investigations should focus on determining the degree to which the residuals satisfy the properties of zero-mean, white, Gaussian statistics in order to determine the adequacy of the reduced-order model [26].

After simulation efforts complete the initial design and tuning of a Kalman filter, another form of testing is typically applied to ensure that the filter will perform well in the job for which it was designed. Consequently, the Kalman filter should be supplied actual data which has been collected from operational ARS or NRS hardware. The only data of this type which has been collected to date was obtained from CIRIS using the LN-15 or LN-39 INS. Although the system errors in the two INS models are believed to be roughly equivalent, the effect of "driving" the LN-93 INS with data which is collected from the LN-39 INS has not been verified prior to this thesis. It is *assumed* that the effect is negligible [17]. This assumption can be verified or refuted in future research.

Another area which seems intriguing, highly valuable, and under developed is that of Kalman filter state reduction techniques. In many cases, a filter construction based

on a truth model has potential observability problems. Generally, such problems can be identified via observability analyses [22]. In fact, such analysis is attempted in this research but the only tool suitable for the task (*MATRIX*) is not capable of calculating the observability Gramian matrix of the dimensions necessary for this thesis. It can be an extremely valuable project to develop *robust* software which is specifically designed to produce a high-accuracy (i.e. double-precision) observability Gramian for high dimensional systems such as the NRS. This would provide an extremely valuable, time-saving and cost-saving tool to future Kalman filter designers who are faced with design considerations which necessitate system dimension reduction.

Appendix A. NRS Error Model State Definitions

The *complete* NRS error model is composed of 96 LN-93 INS error states (including the baro-altimeter states), 26 RRS transponder states, and 30 GPS error states, for a total of 152 states in the *truth* model. Naturally, there are 152 states in the *full-order* filter model as well. The non-zero elements of the F and Q matrices of the LN-93 error model are presented in Appendix B.

The reader should note that several errors exist in the Litton reference [20]. Many such errors are identified in [13]. Others are noted in [19].

A.1 INS and Baro-altimeter Error States

Tables A.1 through A.4 describe the LN-93 error model (93 states) as defined in the Litton LN-93 CDRL [20]. Note that three additional barometric altitude states which were not explicitly included in the Litton model are added for a total of 96 states. A detailed discussion on the need for and development of the additional baro-altimeter states is included in Chapter 3. These additional baro-altimeter states are inserted in the error state vector sequence at numbers 24, 25, and 26. Additionally, the original state 23 δh_v is now labeled δh_H . Consequently, INS error states which were *originally* numbered 24 \rightarrow 93 are now renumbered as 27 \rightarrow 96.

A.2 RRS Transponder Error States

Table A.5 defines the RRS transponder error states as they are modeled in the NRS. These states are defined in and extracted from [31]. A total of 26 states are included to model the error characteristics of six ground transponders plus interrogator error sources.

A.3 GPS Error States

Table A.6 defines the GPS error states as they are modeled in the NRS. These states are defined in and extracted from [11]. The definitions are believed to be ultimately traceable to the paper by D.B. Cox [6]. A total of 30 states are included to model the error characteristics of 4 space vehicles plus user equipment error sources.

Table A.1. NRS System Model: INS States 1 → 22

State Number	State Symbol	Definition
1	$\delta\theta_x$	X-component of vector angle from true to computer frame
2	$\delta\theta_y$	Y-component of vector angle from true to computer frame
3	$\delta\theta_z$	Z-component of vector angle from true to computer frame
4	ϕ_x	X-component of vector angle from true to platform frame
5	ϕ_y	Y-component of vector angle from true to platform frame
6	ϕ_z	Z-component of vector angle from true to platform frame
7	δV_x	X-component of error in computed velocity
8	δV_y	Y-component of error in computed velocity
9	δV_z	Z-component of error in computed velocity
10	δh	Error in vehicle altitude above reference ellipsoid
11	δh_L	Error in lagged inertial altitude
12	δS_3	Error in vertical channel aiding state
13	δS_1	Error in vertical channel aiding state
14	$b_{x,c}$	X-component of gyro correlated drift rate
15	$b_{y,c}$	Y-component of gyro correlated drift rate
16	$b_{z,c}$	Z-component of gyro correlated drift rate
17	$\nabla_{x,c}$	X-component of accelerometer and velocity quantizer correlated noise
18	$\nabla_{y,c}$	Y-component of accelerometer and velocity quantizer correlated noise
19	$\nabla_{z,c}$	Z-component of accelerometer and velocity quantizer correlated noise
20	δg_x	X-component of gravity vector errors
21	δg_y	Y-component of gravity vector errors
22	δg_z	Z-component of gravity vector errors

Table A.2. NRS System Model: INS States 23 → 50

State Number	State Symbol	Definition
23	δh_B	Total baro-altimeter correlated error
24	δh_c	Baro-altimeter correlated noise error
25	δh_b	Baro-altimeter bias error
26	δh_{sf}	Baro-altimeter scale factor error
27	b_{x_t}	X-component of gyro trend
28	b_{y_t}	Y-component of gyro trend
29	b_{z_t}	Z-component of gyro trend
30	∇_{x_t}	X-component of accelerometer trend
31	∇_{y_t}	Y-component of accelerometer trend
32	∇_{z_t}	Z-component of accelerometer trend
33	b_x	X-component of gyro drift rate repeatability
34	b_y	Y-component of gyro drift rate repeatability
35	b_z	Z-component of gyro drift rate repeatability
36	S_{gx}	X-component of gyro scale factor error
37	S_{gy}	Y-component of gyro scale factor error
38	S_{gz}	Z-component of gyro scale factor error
39	χ_1	X gyro misalignment about Y-axis
40	χ_2	Y gyro misalignment about X-axis
41	χ_3	Z gyro misalignment about X-axis
42	ν_1	X gyro misalignment about Z-axis
43	ν_2	Y gyro misalignment about Z-axis
44	ν_3	Z gyro misalignment about Y-axis
45	D_{xxx}	X gyro scale factor non-linearity
46	D_{yyy}	Y gyro scale factor non-linearity
47	D_{zzz}	Z gyro scale factor non-linearity
48	S_{qb_x}	X gyro scale factor asymmetry error
49	S_{qb_y}	Y gyro scale factor asymmetry error
50	S_{qb_z}	Z gyro scale factor asymmetry error

Table A.3. NRS System Model: INS States 51 → 72

State Number	State Symbol	Definition
51	∇_{b_x}	X-component of accelerometer bias repeatability
52	∇_{b_y}	Y-component of accelerometer bias repeatability
53	∇_{b_z}	Z-component of accelerometer bias repeatability
54	S_{A_x}	X-component of accelerometer and velocity quantizer scale factor error
55	S_{A_y}	Y-component of accelerometer and velocity quantizer scale factor error
56	S_{A_z}	Z-component of accelerometer and velocity quantizer scale factor error
57	S_{QA_x}	X-component of accelerometer and velocity quantizer scale factor asymmetry
58	S_{QA_y}	Y-component of accelerometer and velocity quantizer scale factor asymmetry
59	S_{QA_z}	Z-component of accelerometer and velocity quantizer scale factor asymmetry
60	f_{xx}	Coefficient of error proportional to square of measured acceleration
61	f_{yy}	Coefficient of error proportional to square of measured acceleration
62	f_{zz}	Coefficient of error proportional to square of measured acceleration
63	f_{xy}	Coefficient of error proportional to products of acceleration along and orthogonal to accelerometer sensitive axis
64	f_{xz}	Coefficient of error proportional to products of acceleration along and orthogonal to accelerometer sensitive axis
65	f_{yx}	Coefficient of error proportional to products of acceleration along and orthogonal to accelerometer sensitive axis
66	f_{yz}	Coefficient of error proportional to products of acceleration along and orthogonal to accelerometer sensitive axis
67	f_{zx}	Coefficient of error proportional to products of acceleration along and orthogonal to accelerometer sensitive axis
68	f_{zy}	Coefficient of error proportional to products of acceleration along and orthogonal to accelerometer sensitive axis
69	μ_1	X accelerometer misalignment about Z-axis
70	μ_2	Y accelerometer misalignment about Z-axis
71	μ_3	Z accelerometer misalignment about Y-axis
72	σ_3	Z-accelerometer misalignment about X-axis

Table A.4. NRS System Model: INS States 73 → 96

State Number	State Symbol	Definition
73	∇_{xq}	X-component of accelerometer bias thermal transient
74	∇_{yq}	Y-component of accelerometer bias thermal transient
75	∇_{zq}	Z-component of accelerometer bias thermal transient
76	b_{xq}	X-component of initial gyro drift rate bias thermal transient
77	b_{yq}	Y-component of initial gyro drift rate bias thermal transient
78	b_{zq}	Z-component of initial gyro drift rate bias thermal transient
79	F_{xyx}	X gyro compliance term
80	F_{xyy}	X gyro compliance term
81	F_{xyx}	X gyro compliance term
82	F_{xxy}	X gyro compliance term
83	F_{xxz}	X gyro compliance term
84	F_{xxz}	X gyro compliance term
85	F_{yxx}	Y gyro compliance term
86	F_{yxx}	Y gyro compliance term
87	F_{yyx}	Y gyro compliance term
88	F_{yyx}	Y gyro compliance term
89	F_{yxx}	Y gyro compliance term
90	F_{yxy}	Y gyro compliance term
91	F_{xxy}	Z gyro compliance term
92	F_{xxy}	Z gyro compliance term
93	F_{xxy}	Z gyro compliance term
94	F_{xyx}	Z gyro compliance term
95	F_{xyy}	Z gyro compliance term
96	F_{xyx}	Z gyro compliance term

Table A.5. NRS System Model: RRS Error States

State Number	State Symbol	Definition [NOTE: S_{INS} = Total INS States]
$S_{INS} + 1$	δR_b	Range error due to equipment bias
$S_{INS} + 2$	δv_b	Velocity error due to equipment bias
$S_{INS} + 3$	δP_{T1x}	Transponder 1 x-component of position error
$S_{INS} + 4$	δP_{T1y}	Transponder 1 y-component of position error
$S_{INS} + 5$	δP_{T1z}	Transponder 1 z-component of position error
$S_{INS} + 6$	δR_{T1a}	Transponder 1 range error due to atm propagation
$S_{INS} + 7$	δP_{T2x}	Transponder 2 x-component of position error
$S_{INS} + 8$	δP_{T2y}	Transponder 2 y-component of position error
$S_{INS} + 9$	δP_{T2z}	Transponder 2 z-component of position error
$S_{INS} + 10$	δR_{T2a}	Transponder 2 range error due to atm propagation
$S_{INS} + 11$	δP_{T3x}	Transponder 3 x-component of position error
$S_{INS} + 12$	δP_{T3y}	Transponder 3 y-component of position error
$S_{INS} + 13$	δP_{T3z}	Transponder 3 z-component of position error
$S_{INS} + 14$	δR_{T3a}	Transponder 3 range error due to atm propagation
$S_{INS} + 15$	δP_{T4x}	Transponder 4 x-component of position error
$S_{INS} + 16$	δP_{T4y}	Transponder 4 y-component of position error
$S_{INS} + 17$	δP_{T4z}	Transponder 4 z-component of position error
$S_{INS} + 18$	δR_{T4a}	Transponder 4 range error due to atm propagation
$S_{INS} + 19$	δP_{T5x}	Transponder 5 x-component of position error
$S_{INS} + 20$	δP_{T5y}	Transponder 5 y-component of position error
$S_{INS} + 21$	δP_{T5z}	Transponder 5 z-component of position error
$S_{INS} + 22$	δR_{T5a}	Transponder 5 range error due to atm propagation
$S_{INS} + 23$	δP_{T6x}	Transponder 6 x-component of position error
$S_{INS} + 24$	δP_{T6y}	Transponder 6 y-component of position error
$S_{INS} + 25$	δP_{T6z}	Transponder 6 z-component of position error
$S_{INS} + 26$	δR_{T6a}	Transponder 6 range error due to atm propagation

Table A.6. NRS System Model: GPS Error States

State Number	State Symbol	Definition [NOTE: S_{RRS} = Total RRS States]
$S_{INS} + S_{RRS} + 1$	δR_{clk_1}	User clock bias
$S_{INS} + S_{RRS} + 2$	δD_{clk_1}	User clock drift
$S_{INS} + S_{RRS} + 3$	δR_{loop_1}	SV 1 code loop error
$S_{INS} + S_{RRS} + 4$	δR_{trop_1}	SV 1 tropospheric error
$S_{INS} + S_{RRS} + 5$	δR_{ion_1}	SV 1 ionospheric error
$S_{INS} + S_{RRS} + 6$	$\delta R_{clk_{sv1}}$	SV 1 clock error
$S_{INS} + S_{RRS} + 7$	δx_{sv1}	SV 1 x-component of position error
$S_{INS} + S_{RRS} + 8$	δy_{sv1}	SV 1 y-component of position error
$S_{INS} + S_{RRS} + 9$	δz_{sv1}	SV 1 z-component of position error
$S_{INS} + S_{RRS} + 10$	δR_{loop_2}	SV 2 code loop error
$S_{INS} + S_{RRS} + 11$	δR_{trop_2}	SV 2 tropospheric error
$S_{INS} + S_{RRS} + 12$	δR_{ion_2}	SV 2 ionospheric error
$S_{INS} + S_{RRS} + 13$	$\delta R_{clk_{sv2}}$	SV 2 clock error
$S_{INS} + S_{RRS} + 14$	δx_{sv2}	SV 2 x-component of position error
$S_{INS} + S_{RRS} + 15$	δy_{sv2}	SV 2 y-component of position error
$S_{INS} + S_{RRS} + 16$	δz_{sv2}	SV 2 z-component of position error
$S_{INS} + S_{RRS} + 17$	δR_{loop_3}	SV 3 code loop error
$S_{INS} + S_{RRS} + 18$	δR_{trop_3}	SV 3 tropospheric error
$S_{INS} + S_{RRS} + 19$	δR_{ion_3}	SV 3 ionospheric error
$S_{INS} + S_{RRS} + 20$	$\delta R_{clk_{sv3}}$	SV 3 clock error
$S_{INS} + S_{RRS} + 21$	δx_{sv3}	SV 3 x-component of position error
$S_{INS} + S_{RRS} + 22$	δy_{sv3}	SV 3 y-component of position error
$S_{INS} + S_{RRS} + 23$	δz_{sv3}	SV 3 z-component of position error
$S_{INS} + S_{RRS} + 24$	δR_{loop_4}	SV 4 code loop error
$S_{INS} + S_{RRS} + 25$	δR_{trop_4}	SV 4 tropospheric error
$S_{INS} + S_{RRS} + 26$	δR_{ion_4}	SV 4 ionospheric error
$S_{INS} + S_{RRS} + 27$	$\delta R_{clk_{sv4}}$	SV 4 clock error
$S_{INS} + S_{RRS} + 28$	δx_{sv4}	SV 4 x-component of position error
$S_{INS} + S_{RRS} + 29$	δy_{sv4}	SV 4 y-component of position error
$S_{INS} + S_{RRS} + 30$	δz_{sv4}	SV 4 z-component of position error

Appendix B. *Litton LN-99 Error-State Model Dynamics Matrix*

The LN-93 error-state dynamics matrix (F) as provided by Litton is a 93-by-93 array that contains a large number of elements that are identically zero. Litton partitions the F matrix into thirty-six subarrays [20] reflecting the logical divisions of error sources discussed in Chapter III.

The reader should note that only the NON-ZERO elements are included in the tables which follow, and should further note that the revised baro-altimeter model states are NOT included in this set of ORIGINAL F matrix elements extracted from the Litton document [20].

A notational convention [13] is to label elements of the C_s^t , sensor-to-true, matrix as C_{ij} where i is the row and j is the column in the transformation matrix.

Table B.1. Elements of the Dynamics Submatrix F_{11} [20, 13]

Element	Term	Element	Term
(1,3)	$-\rho_y$	(1,8)	$-C_{RY}$
(2,3)	ρ_x	(2,7)	C_{RX}
(3,1)	ρ_y	(3,2)	$-\rho_x$
(4,2)	$-\Omega_z$	(4,3)	Ω_y
(4,5)	ω_{iz}	(4,6)	$-\omega_{iy}$
(4,8)	$-C_{RY}$	(5,1)	Ω_z
(5,3)	$-\Omega_x$	(5,4)	$-\omega_{iz}$
(5,6)	ω_{ix}	(5,7)	C_{RX}
(6,1)	$-\Omega_y$	(6,2)	Ω_x
(6,4)	ω_{iy}	(6,5)	$-\omega_{ix}$
(7,1)	$-2V_y\Omega_y - 2V_z\Omega_z$	(7,2)	$2V_y\Omega_x$
(7,3)	$2V_z\Omega_y$	(7,5)	$-A_z$
(7,6)	A_y	(7,7)	$-V_zC_{RX}$
(7,8)	$2\Omega_z$	(7,9)	$-\rho_y - 2\Omega_y$
(8,1)	$2V_x\Omega_y$	(8,2)	$-2V_x\Omega_x - 2V_z\Omega_z$
(8,3)	$2V_z\Omega_y$	(8,4)	A_z
(8,6)	$-A_x$	(8,7)	$-2\Omega_z$
(8,8)	$-V_zC_{RY}$	(8,9)	$\rho_x + 2\Omega_x$
(9,1)	$2V_x\Omega_z$	(9,2)	$2V_y\Omega_z$
(9,3)	$-2V_y\Omega_y - 2V_x\Omega_x$	(9,4)	$-A_y$
(9,5)	A_x	(9,7)	$\rho_y + 2\Omega_y + V_xC_{RX}$
(9,8)	$-\rho_x - 2\Omega_x + V_yC_{RY}$	(9,10)	$2g_0/a$
(9,11)	$-k_2$	(9,12)	-1
(9,13)	k_2	(10,9)	1
(10,11)	$-k_1$	(10,13)	$k_1 - 1$
(11,10)	1	(11,11)	-1
(12,11)	k_3	(12,13)	$-k_3$
(13,10)	k_1	(13,11)	$-k_1$
(13,13)	$k_1 - 1$		

Table B.2. Elements of the Dynamics Submatrix F_{12} [20, 13]

Element	Term	Element	Term	Element	Term
(4,14)	C_{11}	(4,15)	C_{12}	(4,16)	C_{13}
(4,24)	$C_{11}t$	(4,25)	$C_{12}t$	(4,26)	$C_{13}t$
(5,14)	C_{21}	(5,15)	C_{22}	(5,16)	C_{23}
(5,24)	$C_{21}t$	(5,25)	$C_{22}t$	(5,26)	$C_{23}t$
(6,14)	C_{31}	(6,15)	C_{32}	(6,16)	C_{33}
(6,24)	$C_{31}t$	(6,25)	$C_{32}t$	(6,26)	$C_{33}t$
(7,17)	C_{11}	(7,18)	C_{12}	(7,19)	C_{13}
(7,20)	1	(7,27)	$C_{11}t$	(7,28)	$C_{12}t$
(7,29)	$C_{13}t$	(8,17)	C_{21}	(8,18)	C_{22}
(8,19)	C_{23}	(8,21)	1	(8,27)	$C_{21}t$
(8,28)	$C_{22}t$	(8,29)	$C_{23}t$	(9,17)	C_{31}
(9,18)	C_{32}	(9,19)	C_{33}	(9,22)	1
(9,23)	k_2	(9,27)	$C_{31}t$	(9,28)	$C_{32}t$
(9,29)	$C_{33}t$	(10,23)	k_1	(12,23)	$-k_3$
(13,23)	$k_4/600$				

Table B.3. Elements of the Dynamics Submatrix F_{13} [20, 13]

Element	Term	Element	Term	Element	Term
(4,30)	C_{11}	(4,31)	C_{12}	(4,32)	C_{13}
(4,33)	$C_{11}\omega_{ib_x}$	(4,34)	$C_{12}\omega_{ib_y}$	(4,35)	$C_{13}\omega_{ib_z}$
(4,36)	$C_{11}\omega_{ib_y}$	(4,37)	$-C_{12}\omega_{ib_x}$	(4,38)	$C_{13}\omega_{ib_y}$
(4,39)	$-C_{11}\omega_{ib_z}$	(4,40)	$C_{12}\omega_{ib_x}$	(4,41)	$-C_{13}\omega_{ib_x}$
(4,42)	$C_{11}\omega_{ib_z}^2$	(4,43)	$C_{12}\omega_{ib_y}^2$	(4,44)	$C_{13}\omega_{ib_z}^2$
(4,45)	$0.5C_{11} \omega_{ib_x} $	(4,46)	$0.5C_{12} \omega_{ib_y} $	(4,47)	$0.5C_{13} \omega_{ib_z} $
(5,30)	C_{21}	(5,31)	C_{22}	(5,32)	C_{23}
(5,33)	$C_{21}\omega_{ib_x}$	(5,34)	$C_{22}\omega_{ib_y}$	(5,35)	$C_{23}\omega_{ib_z}$
(5,36)	$C_{21}\omega_{ib_y}$	(5,37)	$-C_{22}\omega_{ib_x}$	(5,38)	$C_{23}\omega_{ib_y}$
(5,39)	$-C_{21}\omega_{ib_z}$	(5,40)	$C_{22}\omega_{ib_x}$	(5,41)	$-C_{23}\omega_{ib_x}$
(5,42)	$C_{21}\omega_{ib_z}^2$	(5,43)	$C_{22}\omega_{ib_y}^2$	(5,44)	$C_{23}\omega_{ib_z}^2$
(5,45)	$0.5C_{21} \omega_{ib_x} $	(5,46)	$0.5C_{22} \omega_{ib_y} $	(5,47)	$0.5C_{23} \omega_{ib_z} $
(6,30)	C_{31}	(6,31)	C_{32}	(6,32)	C_{33}
(6,33)	$C_{31}\omega_{ib_x}$	(6,34)	$C_{32}\omega_{ib_y}$	(6,35)	$C_{33}\omega_{ib_z}$
(6,36)	$C_{31}\omega_{ib_y}$	(6,37)	$-C_{32}\omega_{ib_x}$	(6,38)	$C_{33}\omega_{ib_y}$
(6,39)	$-C_{31}\omega_{ib_z}$	(6,40)	$C_{32}\omega_{ib_x}$	(6,41)	$-C_{33}\omega_{ib_x}$
(6,42)	$C_{31}\omega_{ib_z}^2$	(6,43)	$C_{32}\omega_{ib_y}^2$	(6,44)	$C_{33}\omega_{ib_z}^2$
(6,45)	$0.5C_{31} \omega_{ib_x} $	(6,46)	$0.5C_{32} \omega_{ib_y} $	(6,47)	$0.5C_{33} \omega_{ib_z} $

Table B.4. Elements of the Dynamics Submatrix F_{14} [20, 13]

Element	Term	Element	Term	Element	Term
(7,48)	C_{11}	(7,49)	C_{12}	(7,50)	C_{13}
(7,51)	$C_{11}A_x^B$	(7,52)	$C_{12}A_y^B$	(7,53)	$C_{13}A_z^B$
(7,54)	$C_{11} A_x^B $	(7,55)	$C_{12} A_y^B $	(7,56)	$C_{13} A_z^B $
(7,57)	$C_{11}A_x^{B^2}$	(7,58)	$C_{12}A_y^{B^2}$	(7,59)	$C_{13}A_z^{B^2}$
(7,60)	$C_{11}A_x^B A_y^B$	(7,61)	$C_{11}A_x^B A_z^B$	(7,62)	$C_{12}A_y^B A_z^B$
(7,63)	$C_{12}A_x^B A_y^B$	(7,64)	$C_{12}A_x^B A_z^B$	(7,65)	$C_{13}A_y^B A_z^B$
(7,66)	$C_{11}A_y^B$	(7,67)	$-C_{12}A_x^B$	(7,68)	$C_{13}A_z^B$
(7,69)	$C_{13}A_x^B$	(8,48)	C_{21}	(8,49)	C_{22}
(8,50)	C_{23}	(8,51)	$C_{21}A_x^B$	(8,52)	$C_{22}A_y^B$
(8,53)	$C_{23}A_z^B$	(8,54)	$C_{21} A_x^B $	(8,55)	$C_{22} A_y^B $
(8,56)	$C_{23} A_z^B $	(8,57)	$C_{21}A_x^{B^2}$	(8,58)	$C_{22}A_y^{B^2}$
(8,59)	$C_{23}A_z^{B^2}$	(8,60)	$C_{21}A_x^B A_y^B$	(8,61)	$C_{21}A_x^B A_z^B$
(8,62)	$C_{22}A_y^B A_z^B$	(8,63)	$C_{22}A_x^B A_z^B$	(8,64)	$C_{23}A_x^B A_z^B$
(8,65)	$C_{23}A_y^B A_z^B$	(8,66)	$C_{21}A_y^B$	(8,67)	$-C_{22}A_x^B$
(8,68)	$C_{23}A_y^B$	(8,69)	$C_{23}A_z^B$	(9,48)	C_{31}
(9,49)	C_{32}	(9,50)	C_{33}	(9,51)	$C_{31}A_x^B$
(9,52)	$C_{32}A_y^B$	(9,53)	$C_{33}A_z^B$	(9,54)	$C_{31} A_x^B $
(9,55)	$C_{32} A_y^B $	(9,56)	$C_{33} A_z^B $	(9,57)	$C_{31}A_x^{B^2}$
(9,58)	$C_{32}A_y^{B^2}$	(9,59)	$C_{33}A_z^{B^2}$	(9,60)	$C_{31}A_x^B A_y^B$
(9,61)	$C_{31}A_x^B A_z^B$	(9,62)	$C_{32}A_y^B A_z^B$	(9,63)	$C_{32}A_x^B A_z^B$
(9,64)	$C_{33}A_x^B A_z^B$	(9,65)	$C_{33}A_y^B A_z^B$	(9,66)	$C_{31}A_y^B$
(9,67)	$-C_{32}A_x^B$	(9,68)	$C_{33}A_y^B$	(9,69)	$C_{33}A_z^B$

Table B.5. Elements of the Dynamics Submatrix F_{15} [20, 13]

Element	Term	Element	Term	Element	Term
(4,73)	C_{11}	(4,74)	C_{12}	(4,75)	C_{13}
(5,73)	C_{21}	(5,74)	C_{22}	(5,75)	C_{23}
(6,73)	C_{31}	(6,74)	C_{32}	(6,75)	C_{33}
(7,70)	C_{11}	(7,71)	C_{12}	(7,72)	C_{13}
(8,70)	C_{21}	(8,71)	C_{22}	(8,72)	C_{23}
(9,70)	C_{31}	(9,71)	C_{32}	(9,72)	C_{33}

Table B.6. Elements of the Dynamics Submatrix F_{16} [20, 13]

Element	Term	Element	Term	Element	Term
(4,76)	$C_{11}A_y^H\omega_{ib_z}$	(4,77)	$C_{11}A_z^H\omega_{ib_y}$	(4,78)	$C_{11}A_x^H\omega_{ib_z}$
(4,79)	$C_{11}A_z^H\omega_{ib_y}$	(4,80)	$C_{11}A_y^H\omega_{ib_z}$	(4,81)	$C_{11}A_x^H\omega_{ib_y}$
(4,82)	$C_{12}A_y^H\omega_{ib_z}$	(4,83)	$C_{12}A_z^H\omega_{ib_y}$	(4,84)	$C_{12}A_x^H\omega_{ib_z}$
(4,85)	$C_{12}A_z^H\omega_{ib_y}$	(4,86)	$C_{12}A_y^H\omega_{ib_z}$	(4,87)	$C_{12}A_x^H\omega_{ib_y}$
(4,88)	$C_{13}A_y^H\omega_{ib_z}$	(4,89)	$C_{13}A_z^H\omega_{ib_y}$	(4,90)	$C_{13}A_x^H\omega_{ib_z}$
(4,91)	$C_{13}A_z^H\omega_{ib_y}$	(4,92)	$C_{13}A_y^H\omega_{ib_z}$	(4,93)	$C_{13}A_x^H\omega_{ib_y}$
(5,76)	$C_{21}A_y^H\omega_{ib_z}$	(5,77)	$C_{21}A_z^H\omega_{ib_y}$	(5,78)	$C_{21}A_x^H\omega_{ib_z}$
(5,79)	$C_{21}A_z^H\omega_{ib_y}$	(5,80)	$C_{21}A_y^H\omega_{ib_z}$	(5,81)	$C_{21}A_x^H\omega_{ib_y}$
(5,82)	$C_{22}A_y^H\omega_{ib_z}$	(5,83)	$C_{22}A_z^H\omega_{ib_y}$	(5,84)	$C_{22}A_x^H\omega_{ib_z}$
(5,85)	$C_{22}A_z^H\omega_{ib_y}$	(5,86)	$C_{22}A_y^H\omega_{ib_z}$	(5,87)	$C_{22}A_x^H\omega_{ib_y}$
(5,88)	$C_{23}A_y^H\omega_{ib_z}$	(5,89)	$C_{23}A_z^H\omega_{ib_y}$	(5,90)	$C_{23}A_x^H\omega_{ib_z}$
(5,91)	$C_{23}A_z^H\omega_{ib_y}$	(5,92)	$C_{23}A_y^H\omega_{ib_z}$	(5,93)	$C_{23}A_x^H\omega_{ib_y}$
(6,76)	$C_{31}A_y^H\omega_{ib_z}$	(6,77)	$C_{31}A_z^H\omega_{ib_y}$	(6,78)	$C_{31}A_x^H\omega_{ib_z}$
(6,79)	$C_{31}A_z^H\omega_{ib_y}$	(6,80)	$C_{31}A_y^H\omega_{ib_z}$	(6,81)	$C_{31}A_x^H\omega_{ib_y}$
(6,82)	$C_{32}A_y^H\omega_{ib_z}$	(6,83)	$C_{32}A_z^H\omega_{ib_y}$	(6,84)	$C_{32}A_x^H\omega_{ib_z}$
(6,85)	$C_{32}A_z^H\omega_{ib_y}$	(6,86)	$C_{32}A_y^H\omega_{ib_z}$	(6,87)	$C_{32}A_x^H\omega_{ib_y}$
(6,88)	$C_{33}A_y^H\omega_{ib_z}$	(6,89)	$C_{33}A_z^H\omega_{ib_y}$	(6,90)	$C_{33}A_x^H\omega_{ib_z}$
(6,91)	$C_{33}A_z^H\omega_{ib_y}$	(6,92)	$C_{33}A_y^H\omega_{ib_z}$	(6,93)	$C_{33}A_x^H\omega_{ib_y}$

Table B.7. Elements of the Dynamics Submatrix F_{22} [20, 13]

Element	Term	Element	Term	Element	Term
(14,14)	$-\beta_{b_{ga}}$	(15,15)	$-\beta_{b_{ya}}$	(16,16)	$-\beta_{b_{za}}$
(17,17)	$-\beta_{\tau_{ga}}$	(18,18)	$-\beta_{\tau_{ya}}$	(19,19)	$-\beta_{\tau_{za}}$
(20,20)	$-\beta_{h_{ga}}$	(21,21)	$-\beta_{h_{ya}}$	(22,22)	$-\beta_{h_{za}}$
(23,23)	$-\beta_{h_{ga}}$				

Table B.8. Elements of the Dynamics Submatrix F_{25} [20, 13]

Element	Term	Element	Term	Element	Term
(70,70)	$-\beta_{\tau_{ya}}$	(71,71)	$-\beta_{\tau_{za}}$	(72,72)	$-\beta_{\tau_{ya}}$
(73,73)	$-\beta_{b_{ya}}$	(74,74)	$-\beta_{b_{za}}$	(75,75)	$-\beta_{b_{ya}}$

B.1 Elements of the Process Noise Matrix

The Litton document [20] includes a 93-by-93 process noise matrix (**Q**) for the LN-93 error model. Like the **F** matrix, the **Q** matrix is partitioned into subarrays which correspond to the error-state subvectors discussed in Chapter III. The vast majority of the elements in **Q** are identically zero. Only the non-zero elements of **Q** are shown below.

Table B.9. Non-zero Elements of Process Noise Submatrix **Q**₁₁ [20, 13]

Element	Term	Element	Term
(4,4)	$\sigma_{\eta_{b_x}}^2$	(5,5)	$\sigma_{\eta_{b_y}}^2$
(6,6)	$\sigma_{\eta_{b_z}}^2$	(7,7)	$\sigma_{\eta_{l_{1,x}}^2}$
(8,8)	$\sigma_{\eta_{l_{1,y}}^2}$	(9,9)	$\sigma_{\eta_{l_{1,z}}^2}$

Table B.10. Non-zero Elements of Process Noise Submatrix **Q**₂₂ [20, 13]

Element	Term	Element	Term
(14,14)	$2\beta_{b_{x,c}} \sigma_{b_{x,c}}^2$	(15,15)	$2\beta_{b_{y,c}} \sigma_{b_{y,c}}^2$
(16,16)	$2\beta_{b_{z,c}} \sigma_{b_{z,c}}^2$	(17,17)	$2\beta_{\nabla_{x,c}} \sigma_{\nabla_{x,c}}^2$
(18,18)	$2\beta_{\nabla_{y,c}} \sigma_{\nabla_{y,c}}^2$	(18,18)	$2\beta_{\nabla_{x,c}} \sigma_{\nabla_{y,c}}^2$
(20,20)	$2\beta_{\delta_{K_x}} \sigma_{\delta_{K_x}}^2$	(21,21)	$2\beta_{\delta_{K_y}} \sigma_{\delta_{K_y}}^2$
(22,22)	$2\beta_{\delta_{K_z}} \sigma_{\delta_{K_z}}^2$	(23,23)	$2\beta_{\delta_{h_x}} \sigma_{\delta_{h_x}}^2$

Appendix C. 99-State INS Alignment and Flight Simulations

All plots contained in this and subsequent appendices are discussed in Chapter VI (Results). With the exception of Appendix F, all plots contained in this and subsequent appendices contain five traces. The innermost trace (---) on each data plot is the mean error time history for the applicable state and is defined by [22]:

$$\bar{M}_n(t_i) = \frac{1}{N} \sum_{j=1}^N e_j(t_i) = \frac{1}{N} \sum_{j=1}^N \{ \hat{x}_j(t_i) - x_{truej}(t_i) \} \quad (C.1)$$

where $\hat{x}_j(t_i)$ is the filter-computed estimate of variable i and $x_{truej}(t_i)$ is the *truth* model value of the same variable, at time t_i , for sample j , and N is the number of time histories in the simulation (10 in this thesis).

In addition to the center trace, two more pairs of traces are plotted. The first pair (represented by ...) is symmetrically displaced about the mean and as a result follows the "undulations" of the $\hat{M}_n(t_i)$. The locus of these traces is calculated from [23] $\hat{M}_n(t_i) \pm \sqrt{PE(t_i)}$, where $PE(t_i)$ is the *true* error covariance at time t_i . The true standard deviation is calculated from [22]:

$$\sigma_{true}(t_i) = \sqrt{PE(t_i)} = \sqrt{\frac{1}{N-1} \sum_{j=1}^N e_j^2(t_i) - \frac{1}{N-1} \hat{M}_n^2(t_i)} \quad (C.2)$$

where N is the number of runs in the Monte Carlo simulation (10 in this thesis), and $\hat{M}_n^2(t_i)$ is the mean-squared value of the variable at each time of interest (such as measurement times).

The last pair of traces (—) represents the filter computed $\pm \sigma_{filter}$ values for the same variables of interest and are symmetrically displaced about zero because the filter "believes" that it is producing zero mean errors [24]. These quantities are propagated and updated in the MSOFE [5] software using the covariance propagation equation shown in Chapter II. These traces represent the filter's estimate of its own error.

C.1 33-State Model: Alignment Using Litton Initial Conditions

The plots in this section represent results of a 10-run Monte Carlo alignment simulation. In these alignments, INS aiding consists of velocity measurements plus baro-altimeter measurements.

Note that the plot for state 13, ΔS_4 is zero for all time during all alignment simulations. This is a normal condition due to the variable's dependence on altitude rate. The state becomes non-zero during flight runs. All other plots are discussed in Chapter VI (Results).

The filter computed error estimates $[\sigma_{filter}]$ are compared to similar plots contained in the Litton reference documentation [20]. Comparisons are qualitative only and are intended to demonstrate trends. Such comparisons are contained in Chapter VI (Results).

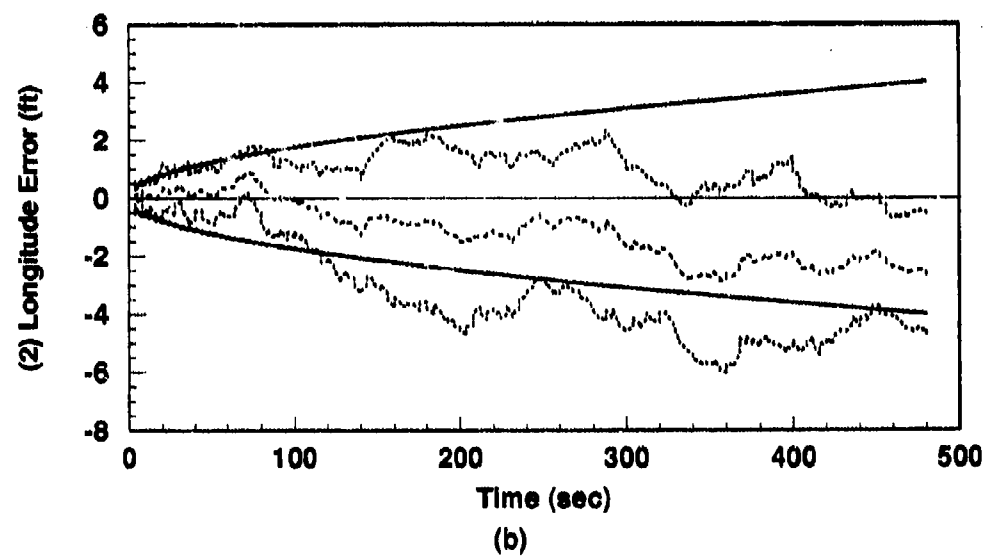
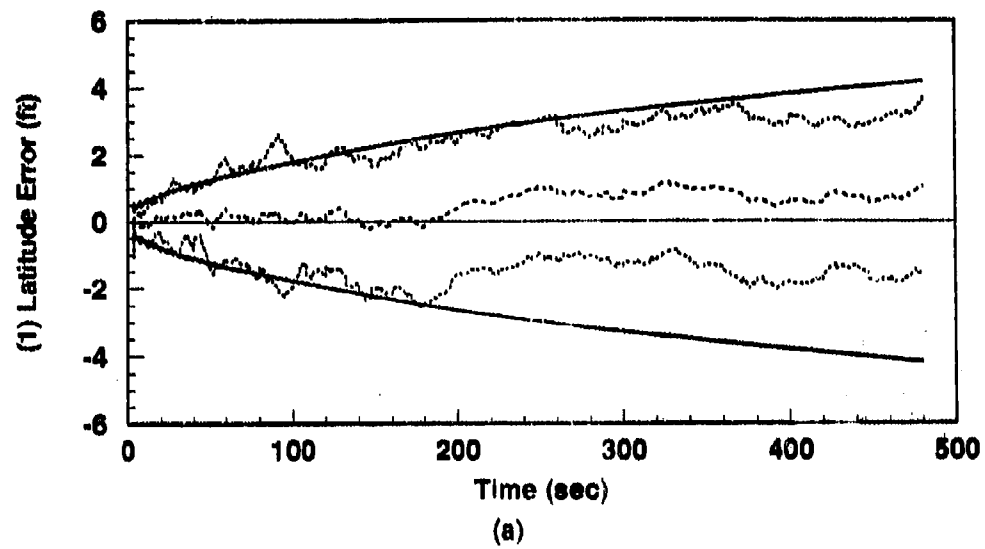
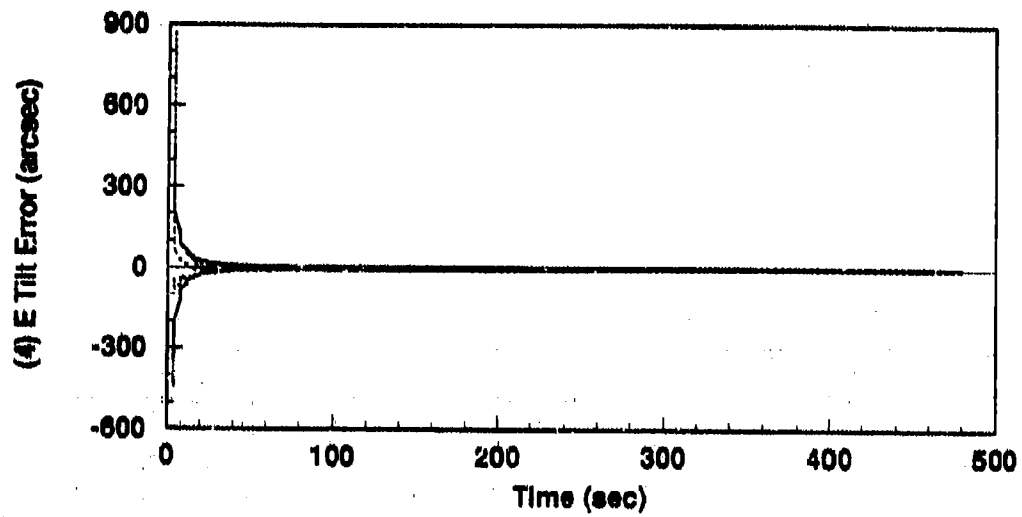
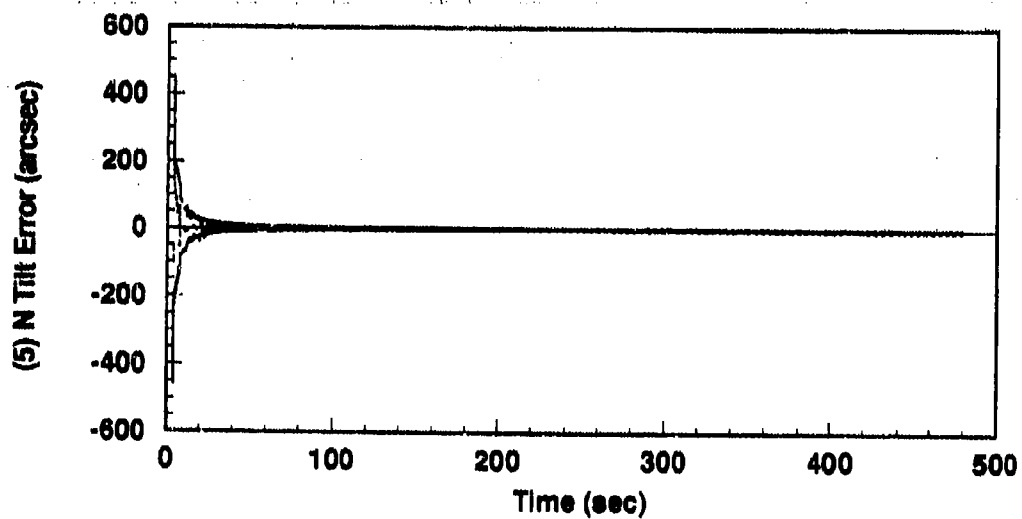


Figure C.1. Alignment: 93-State Model (a) Latitude and (b) Longitude Errors.

----	Mean Error = $\hat{M}_r - (M_r)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$



(a)



(b)

Figure C.2. Alignment: 93-State Model (a) East Tilt and (b) North Tilt Error States.

----	Mean Error = $\bar{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

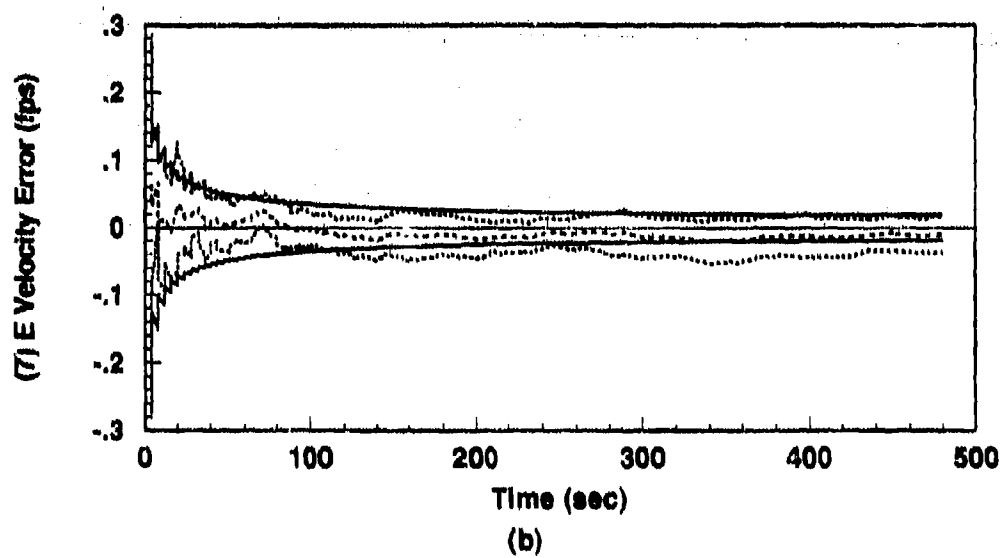
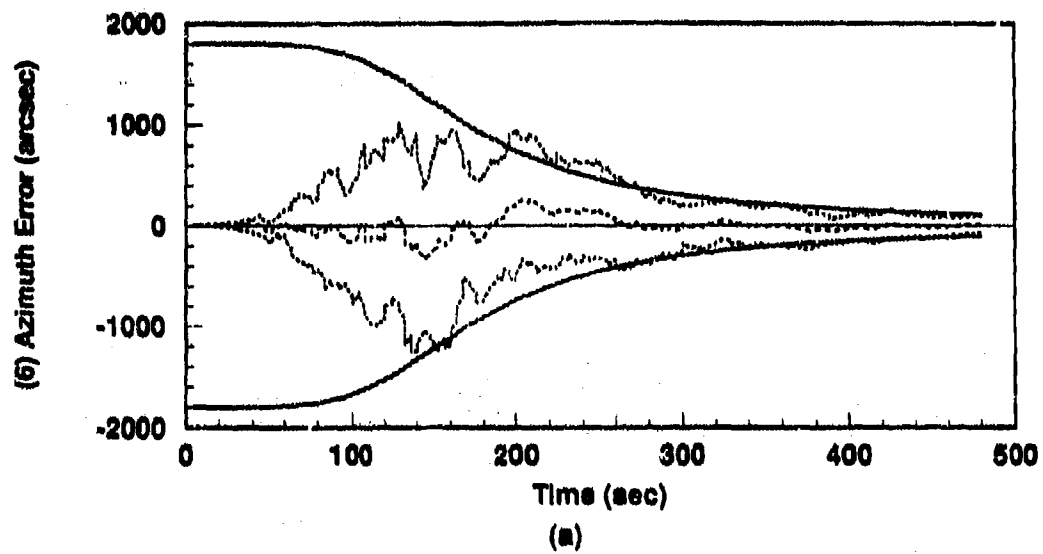


Figure C.3. Alignment: 93-State Model (a) Azimuth and (b) East Velocity Error States.

----	Mean Error = $\bar{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
———	$0 \pm \sigma_{filter}$

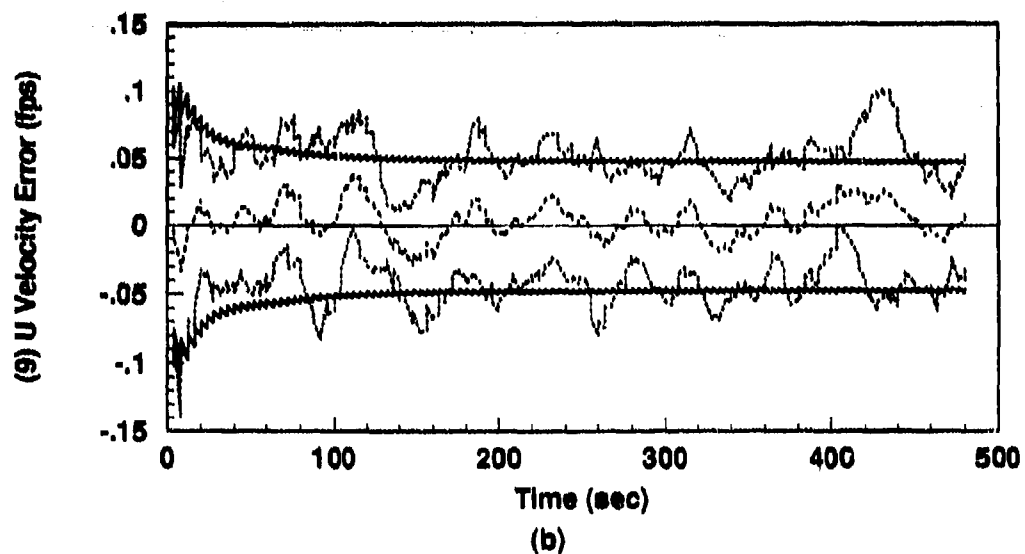
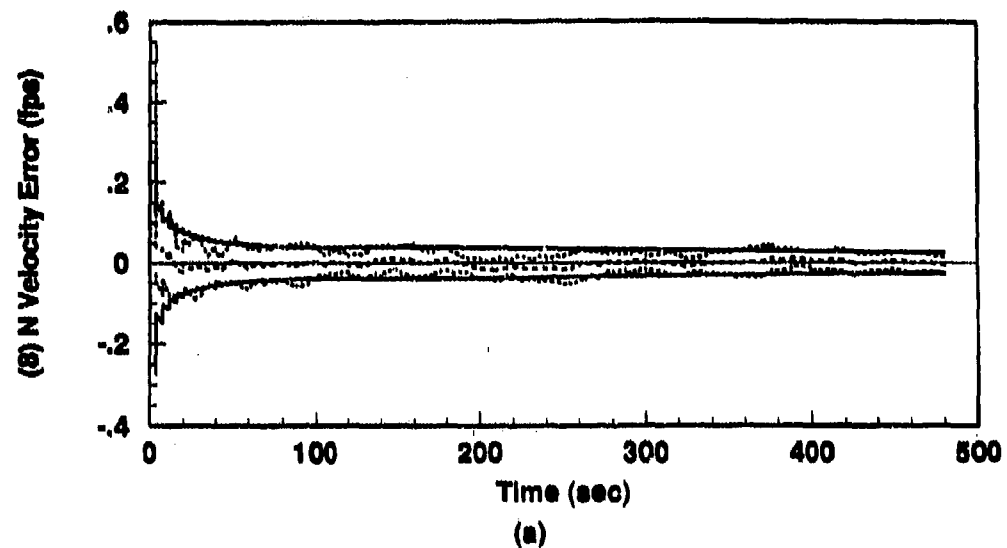


Figure C.4. Alignment: 93-State Model (a) North Velocity and (b) Vertical Velocity Error States.

----	Mean Error = $\hat{M}_1 - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

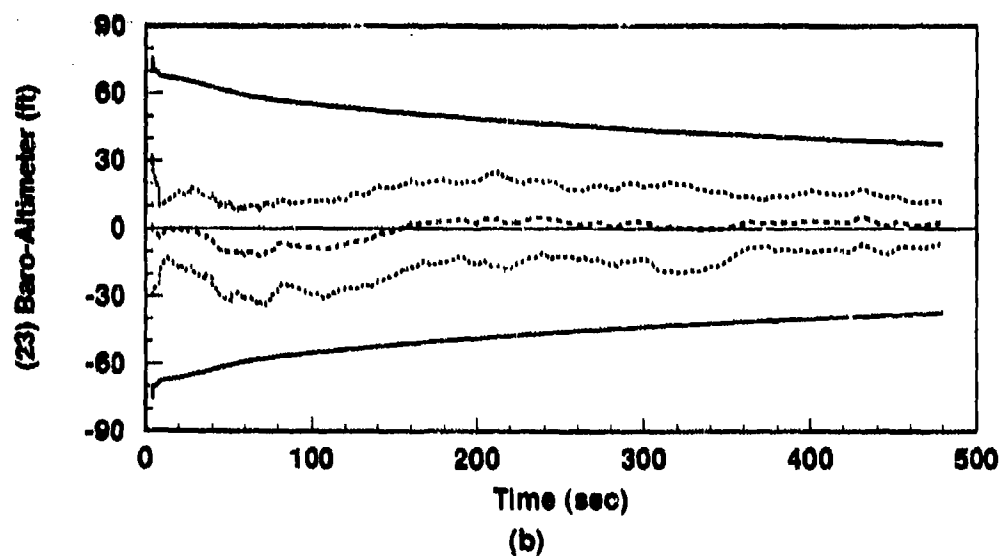
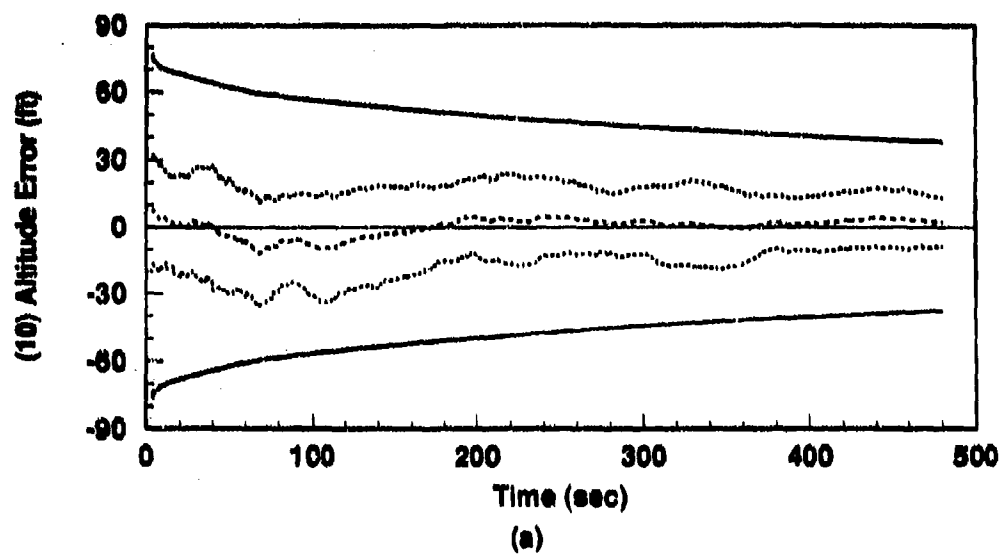
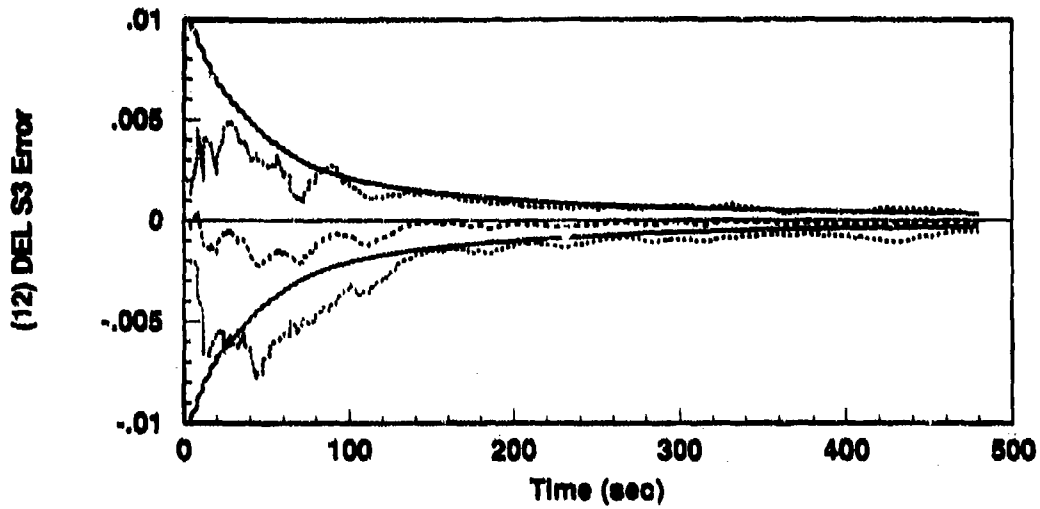
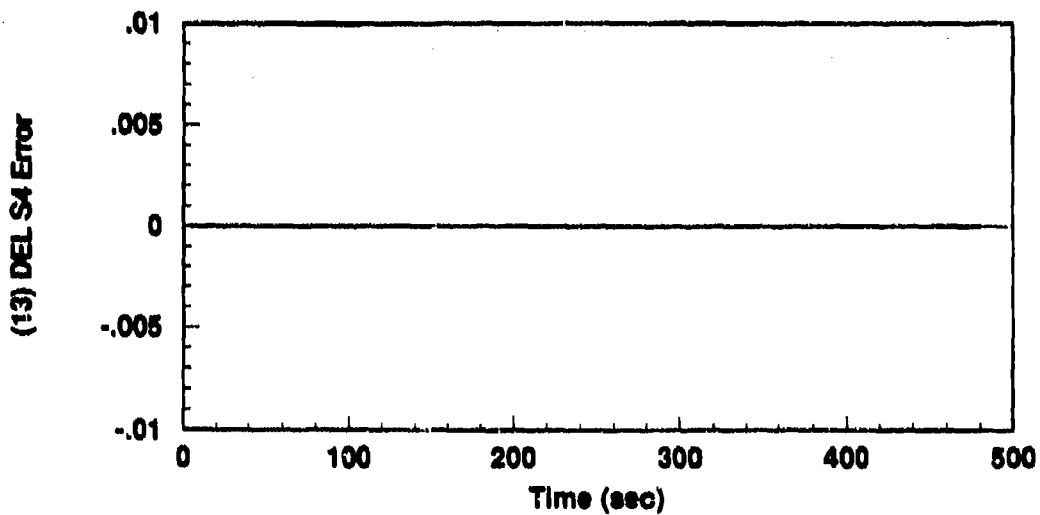


Figure C.5. Alignment: 93-State Model (a) INS Altitude Error State and (b) Baro-Altimeter Total Error.

----	Mean Error = $\bar{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$



(a)



(b)

Figure C.6. Alignment: 93-State Model (a) ΔS_3 and (b) ΔS_4 Vertical Channel Aiding Error States.

----	Mean Error = $\bar{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

C.2 93 State Model: Fighter Flight Using Litton Initial Conditions

The plots in this section represent results of a 10-run flight simulation in which a flight profile (as described in Chapter III) is used to characterize the LN-93 performance for a typical fighter mission. (A discussion of the Litton flight profile is provided in Chapter III.) For this group of runs, baro-altimeter aiding is the only measurement used.

The purpose of this set of runs is to establish that the software implementation is consistent with the Litton error model and that performance results are similar to those obtained by Litton. Consistency in these respects does not constitute error model validation, but does lend confidence that the software functions as intended by Litton.

The filter computed error estimates [σ_{filter}] are compared to similar plots contained in the Litton reference documentation [20]. Comparisons are qualitative only and are intended to demonstrate trends. Such comparisons are contained in Chapter VI (Results).

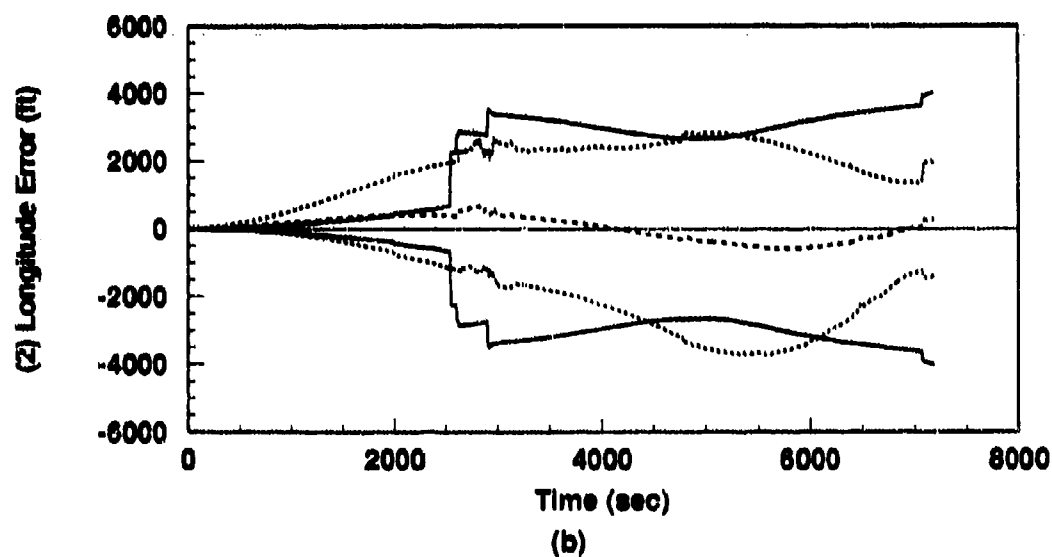
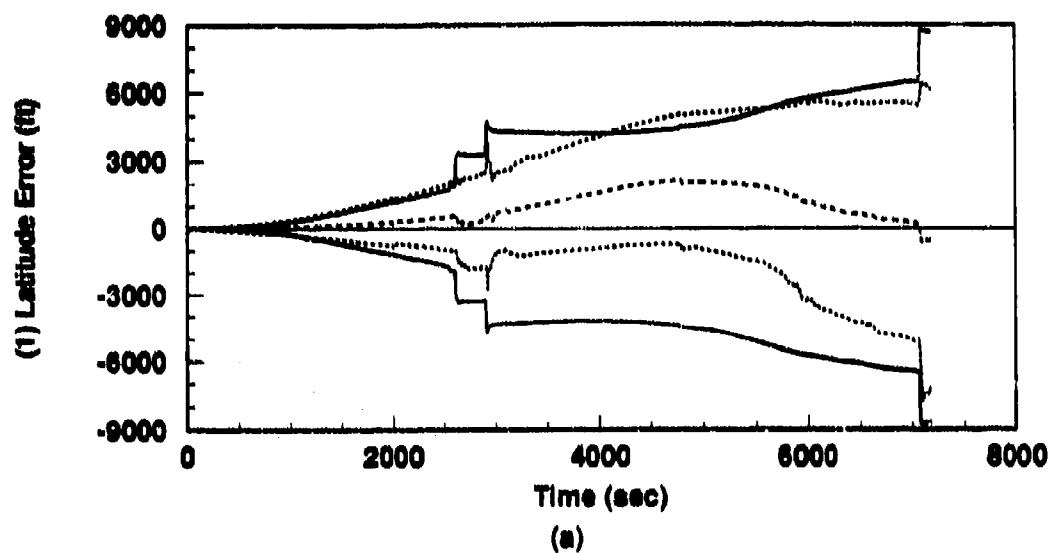


Figure C.7. Flight: 93-State Model (a) Latitude and (b) Longitude Error States.

----	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

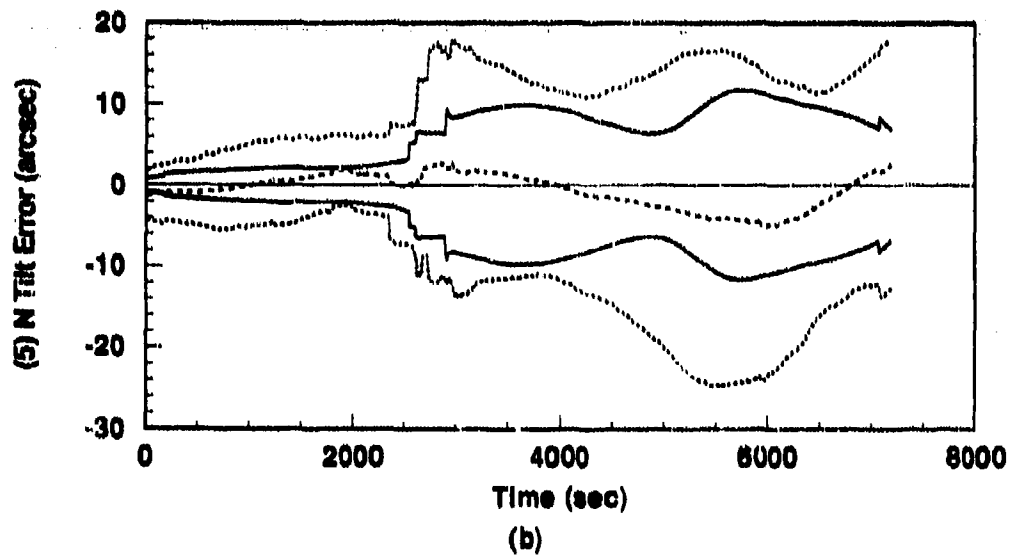
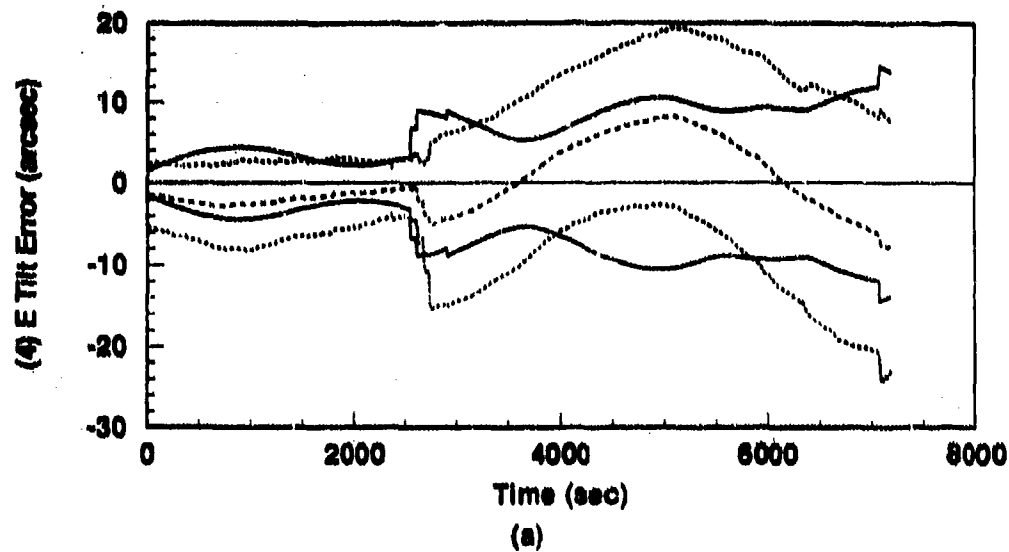


Figure C.8. Flight: 93-State Model (a) East Tilt and (b) North Tilt Error States.

----	Mean Error = $\bar{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

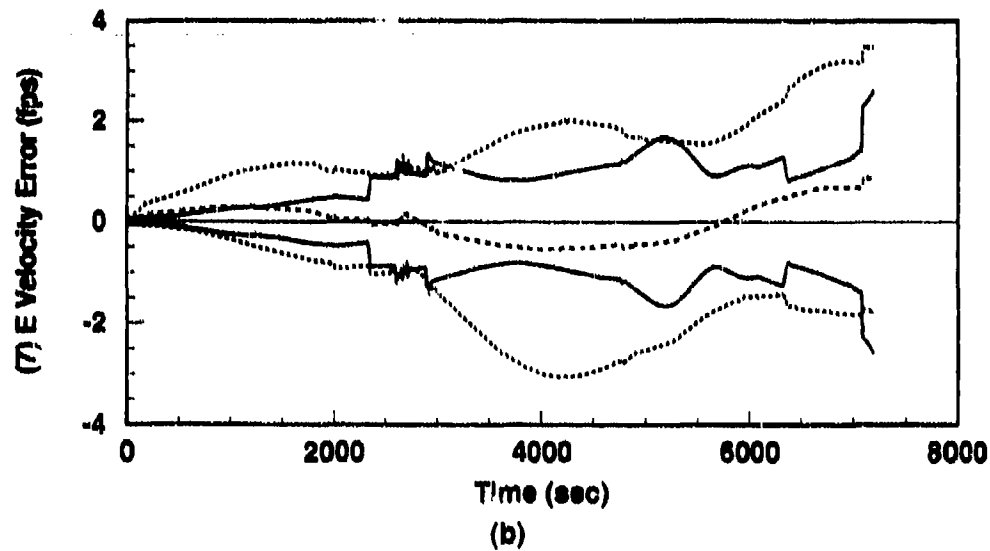
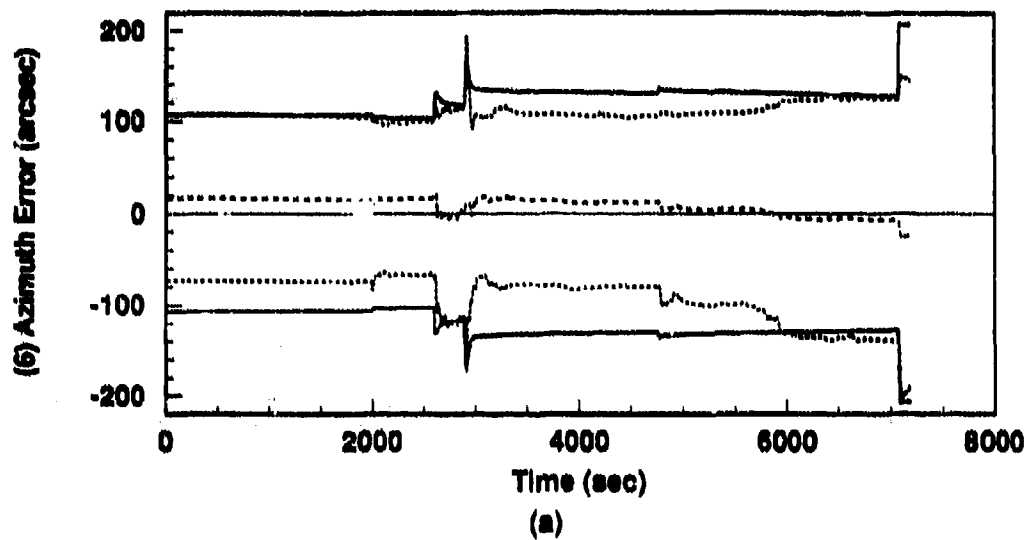


Figure C.9. Flight: 93-State Model (a) Azimuth and (b) East Velocity Error States.

----	Mean Error = $\bar{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

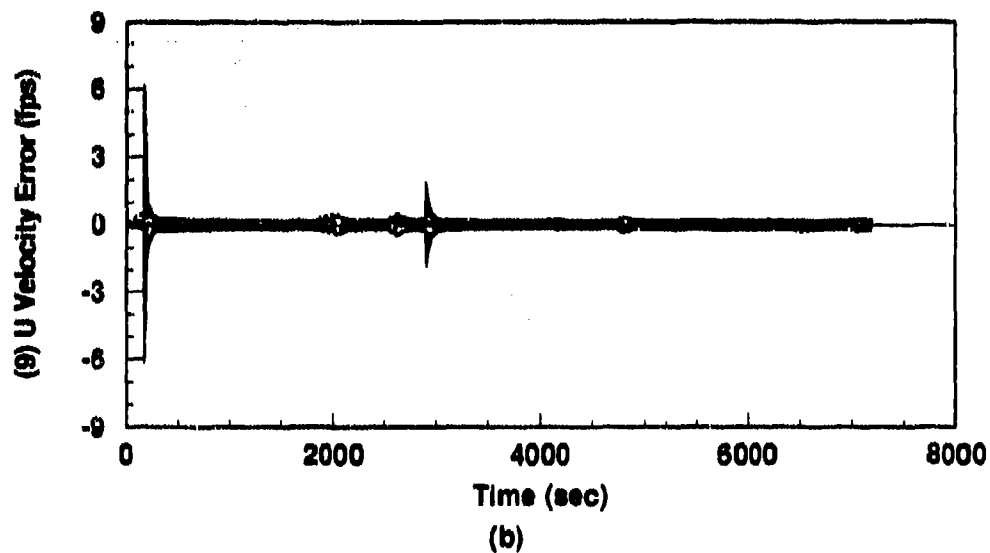
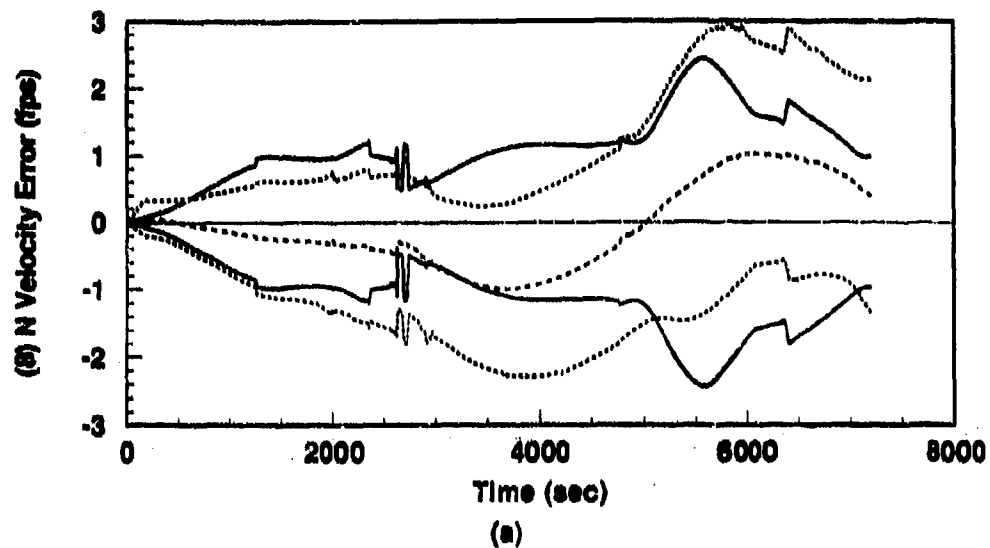


Figure C.10. Flight: 93-State Model (a) North Velocity and (b) Vertical Velocity Error States.

----	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
----	$0 \pm \sigma_{filter}$

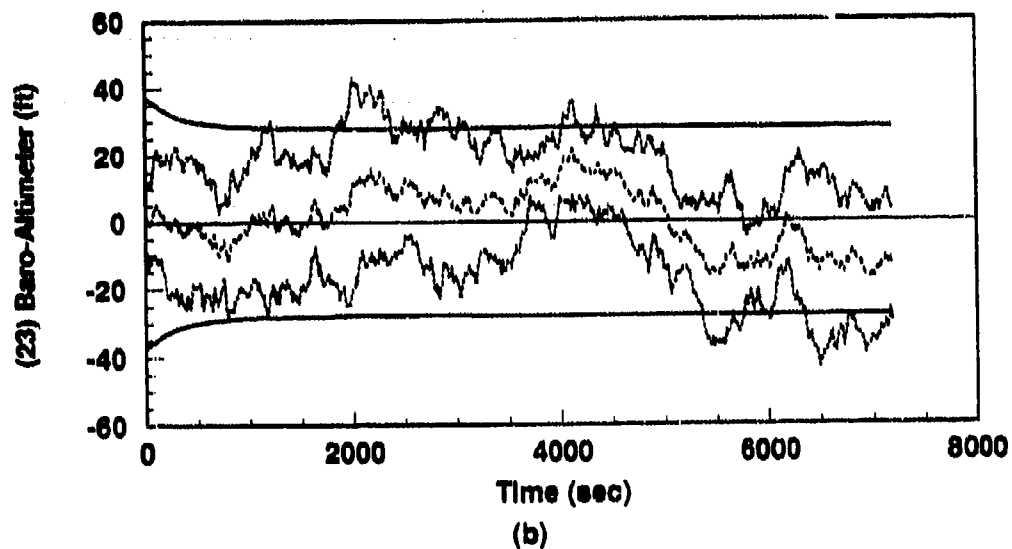
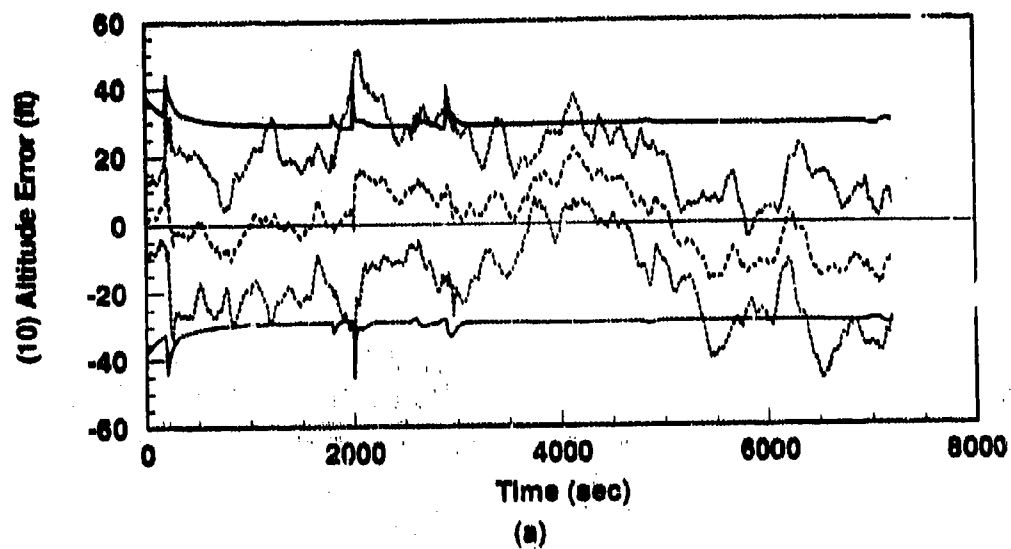


Figure C.11. Flight: 93-State Model (a) INS Altitude Error State and (b) Baro-Altimeter Total Error.

----	Mean Error = $\bar{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
———	$0 \pm \sigma_{filter}$

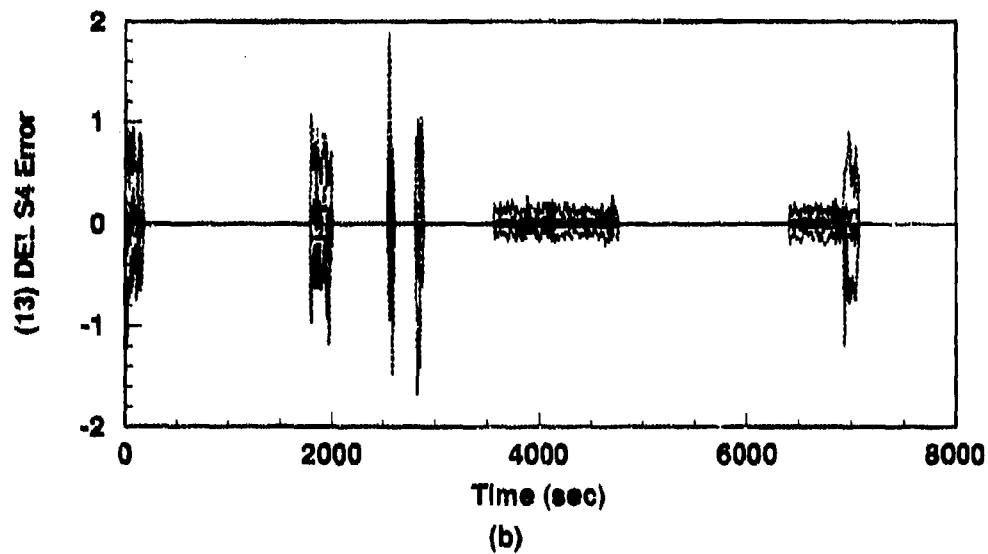
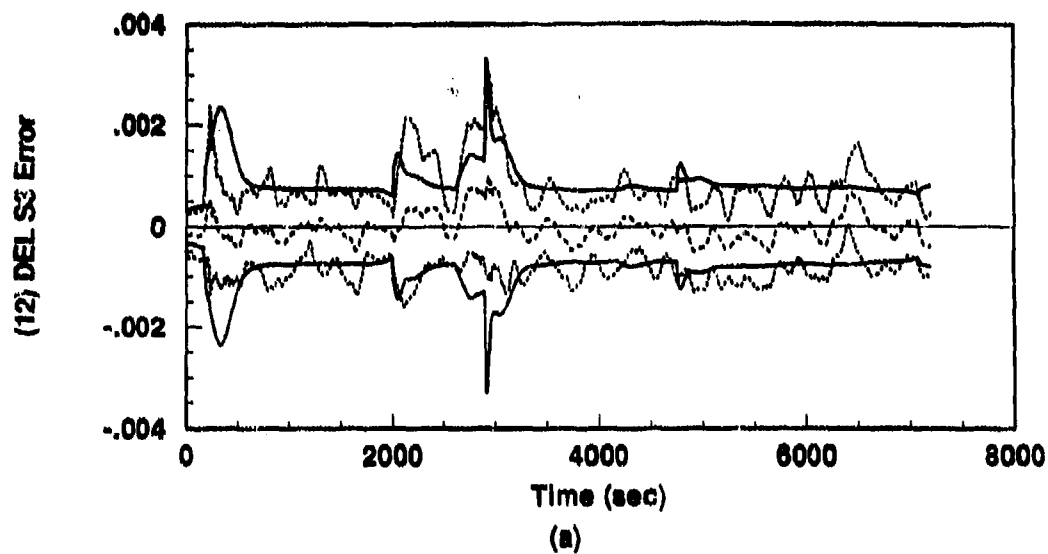


Figure C.12. Flight: 93-State Model (a) ΔS_3 and (b) ΔS_4 Vertical Channel Aiding Error States.

----	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

Appendix D. 96-State INS Alignment and Flight Simulations

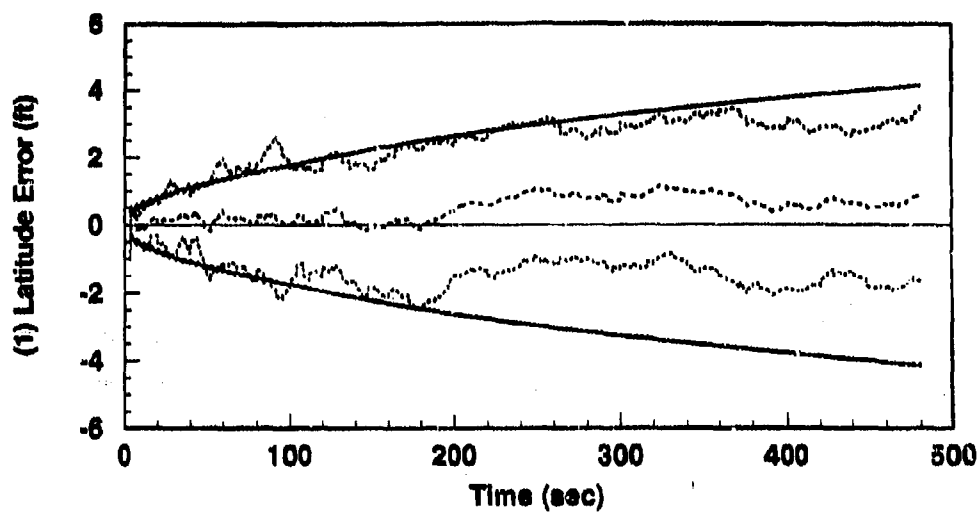
This appendix contains the plots from two sets of 10-run Monte Carlo simulations of the 96-state error model which incorporates the revised baro-altimeter model. Plots contained in this appendix are defined in exactly the same manner as discussed in Appendix C.

D.1 96-State Model: Alignment Using Litton Initial Conditions

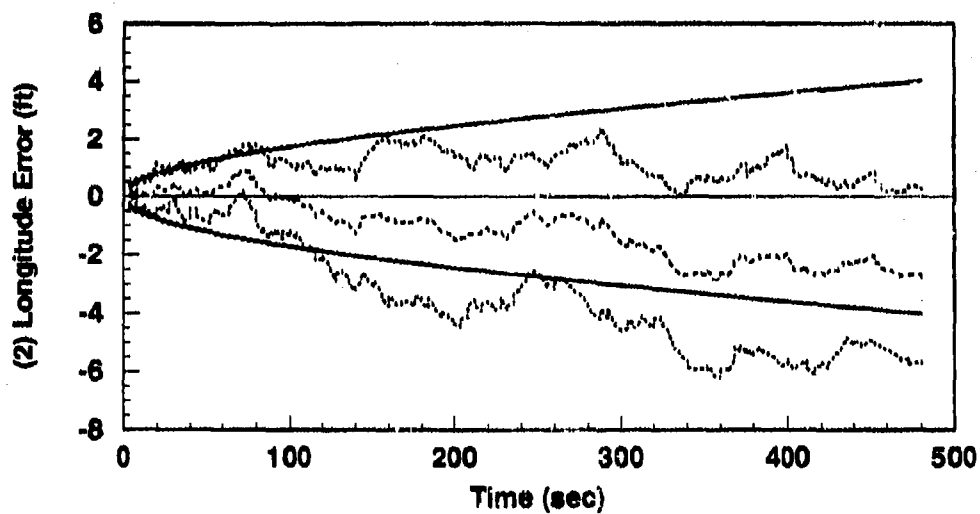
The plots in this section represent results of a 10-run Monte Carlo alignment simulation. In these alignments, INS aiding consists of velocity measurements plus baro-altimeter measurements. Initial conditions are again those chosen by Litton.

Note that the plot for state 13, ΔS_4 , is zero for all time during all alignment simulations. This is a normal condition due to the variable's dependence on altitude rate. The state becomes non-zero during flight runs. All other plots are discussed in Chapter VI (Results).

The filter computed error estimates $[\sigma_{filter}]$ are compared to similar plots contained in the Litton reference documentation [20]. Comparisons are qualitative only and are intended to demonstrate trends. Such comparisons are contained in Chapter VI (Results).



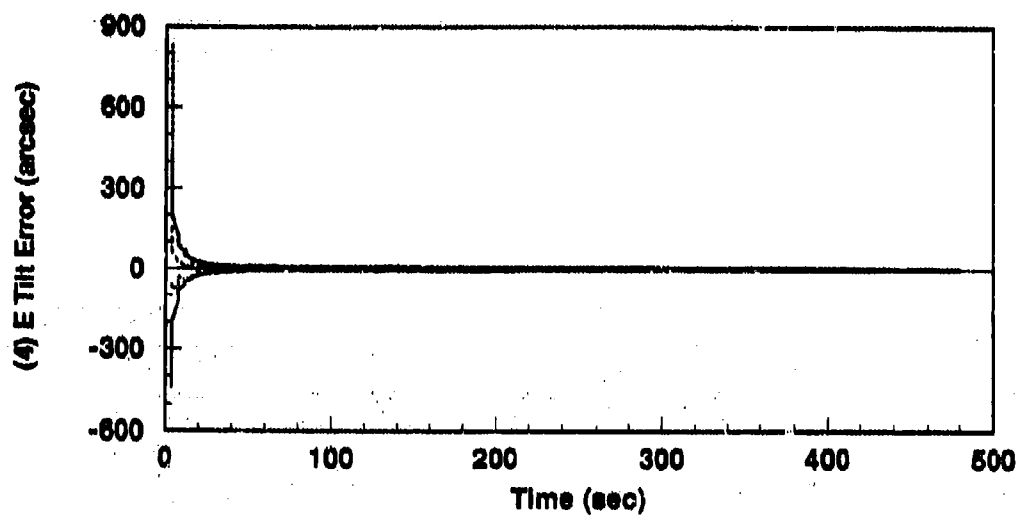
(a)



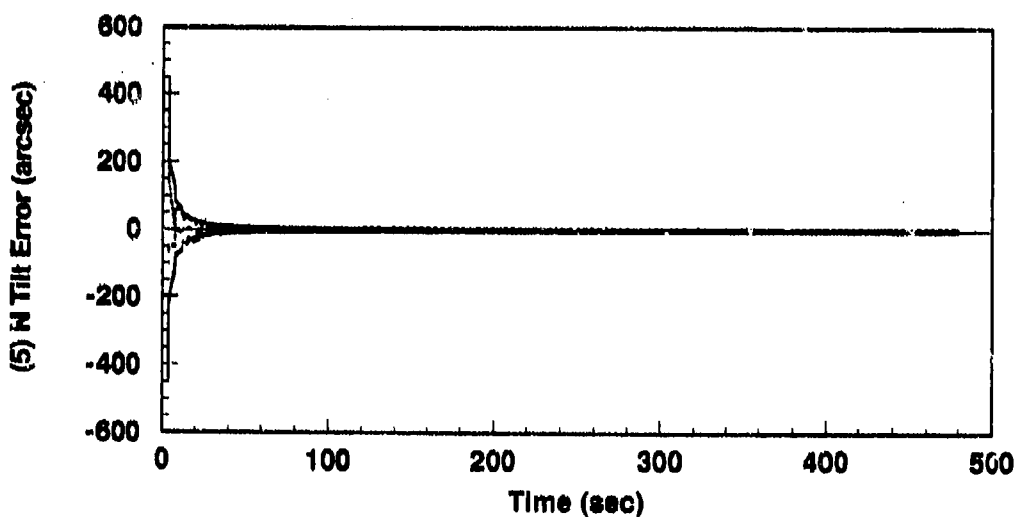
(b)

Figure D.1. Alignment: 96-State Model (a) Latitude and (b) Longitude Errors. (Note: Initial conditions are those chosen by Litton.)

----	Mean Error = $\bar{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$



(a)



(b)

Figure D.2. Alignment: 96-State Model (a) East Tilt and (b) North Tilt Error States.
(Note: Initial conditions are those chosen by Litton.)

----	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

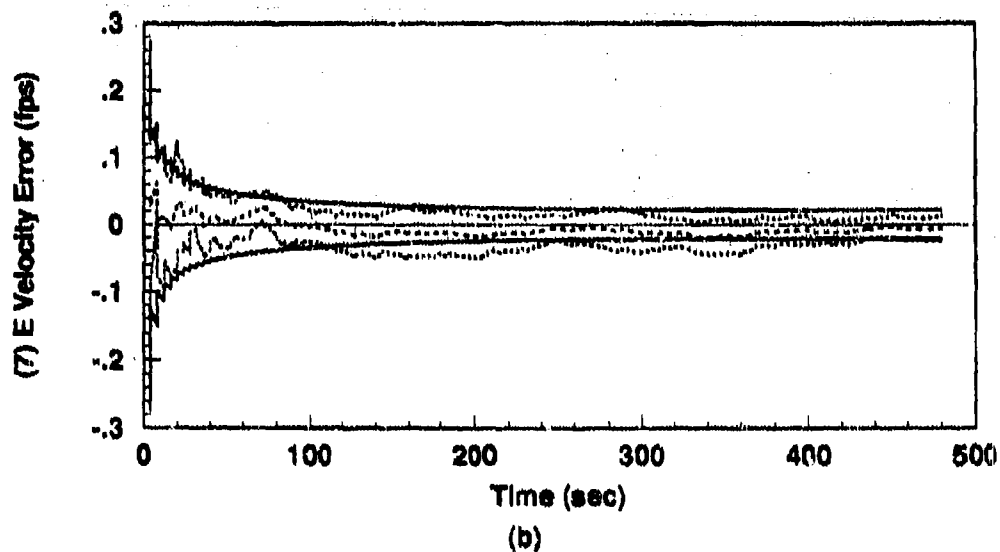
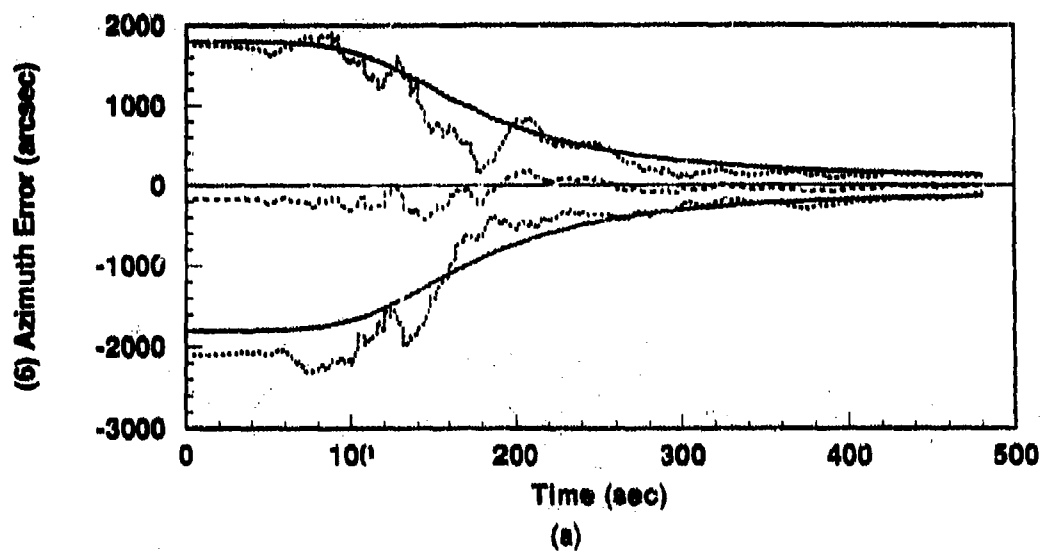


Figure D.3. Alignment: 96-State Model (a) Azimuth and (b) East Velocity Error States.
(Note: Initial conditions are those chosen by Litton.)

----	Mean Error = $\bar{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

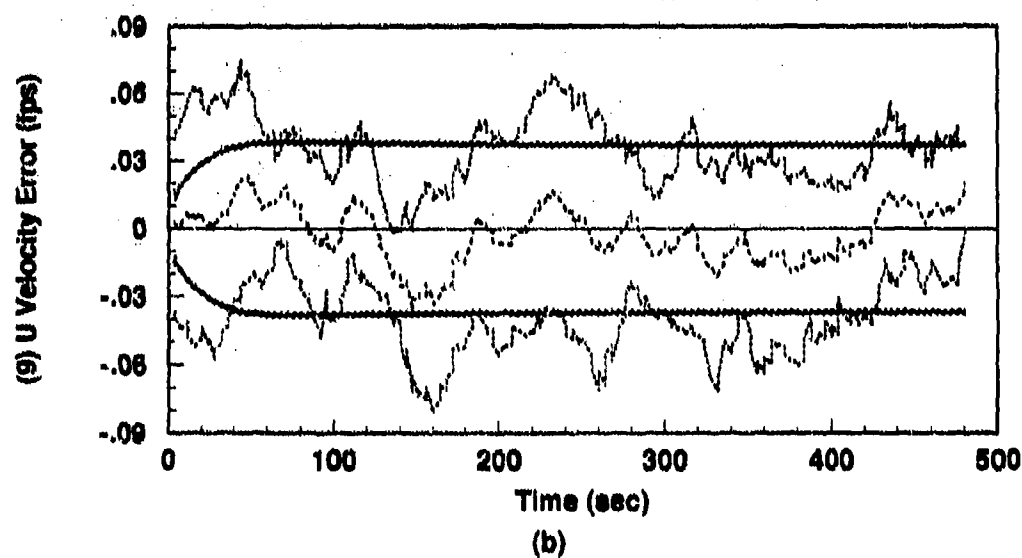
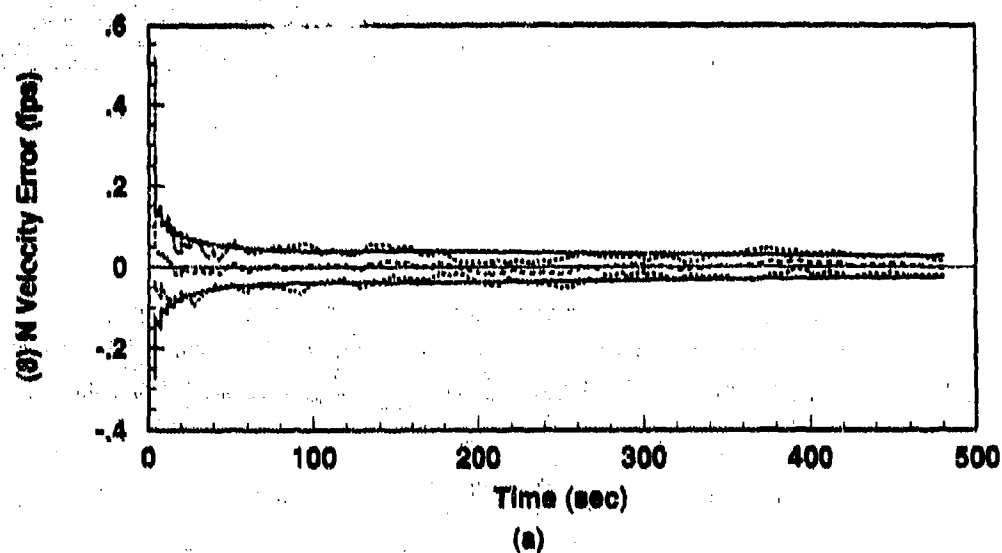


Figure D.4. Alignment: 96-State Model (a) North Velocity and (b) Vertical Velocity Error States. (Note: Initial conditions are those chosen by Litton.)

----	$Mean\ Error = \bar{M}_x - (M_x)_{true}$
.....	$Mean\ Error \pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

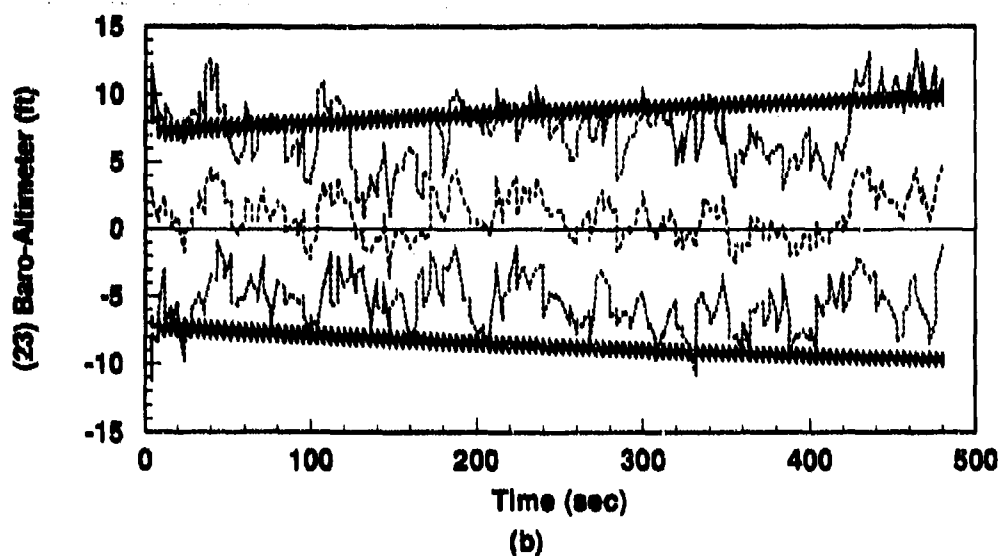
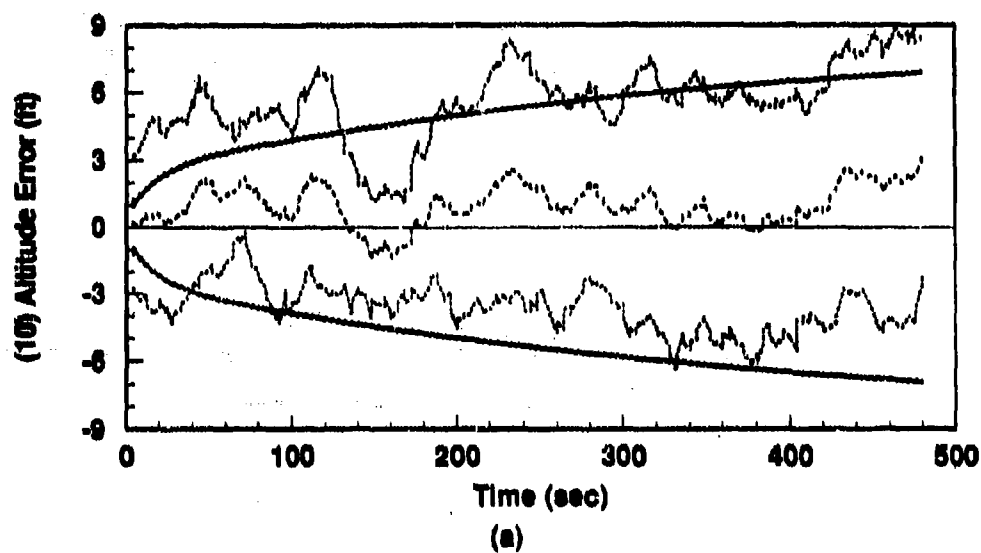


Figure D.5. Alignment: 96-State Model (a) INS Altitude Error State and (b) Baro-Altimeter Total Error. (Note: Initial conditions are those chosen by Litton.)

----	Mean Error = $\bar{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

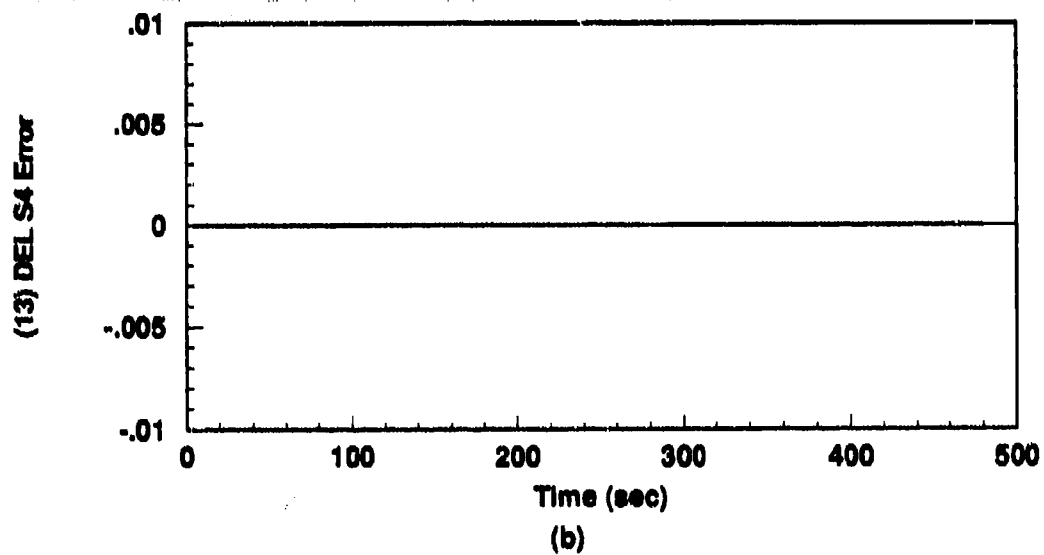
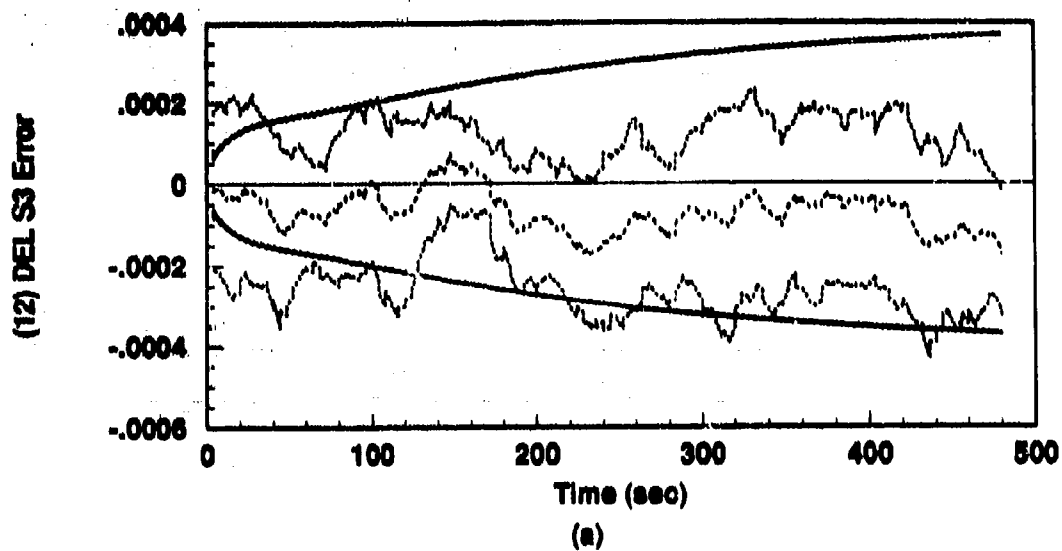


Figure D.6. Alignment: 96-State Model (a) ΔS_3 and (b) ΔS_4 Vertical Channel Aiding Error States. (Note: Initial conditions are those chosen by Litton.)

....	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

D.2 98 State Model: Fighter Flight Using Litton Initial Conditions

The plots in this section represent results of a 10-run flight simulation in which a flight profile (as described in Chapter III) is used to characterize the LN-93 performance for a typical fighter mission. For this group of runs, only baro-altimeter aiding is used. Note, however, that a revised baro-altimeter model is in place in the truth model (see discussion in Chapter III), which accounts for the increase of 3 states in the overall error model.

The purpose of this set of runs is to establish that the software function is not adversely affected by the addition of the new baro-altimeter states.

The filter computed error estimates [σ_{filter}] are compared to similar plots contained in the Litton reference documentation [20]. Comparisons are qualitative only and are intended to demonstrate trends. Such comparisons are contained in Chapter VI (Results).

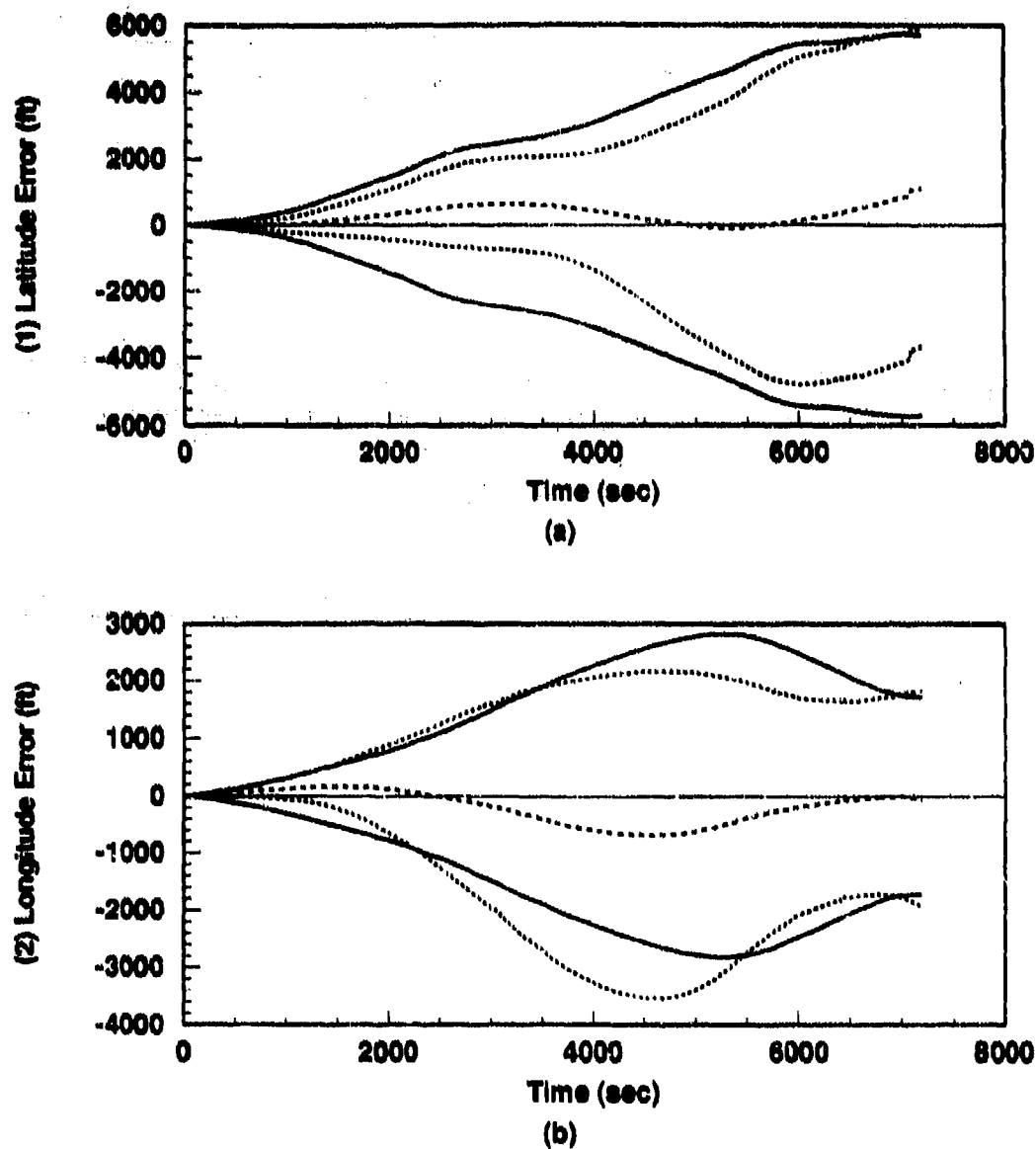


Figure D.7. Flight: 96-State Model (a) Latitude and (b) Longitude Error States. (Note: Initial conditions are those chosen by Litton.)

.....	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
———	$0 \pm \sigma_{filter}$

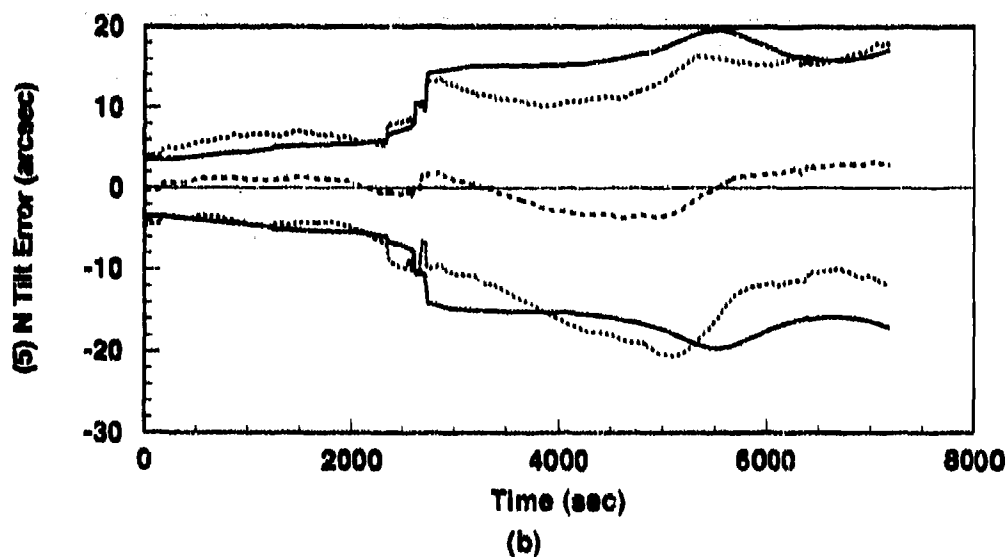
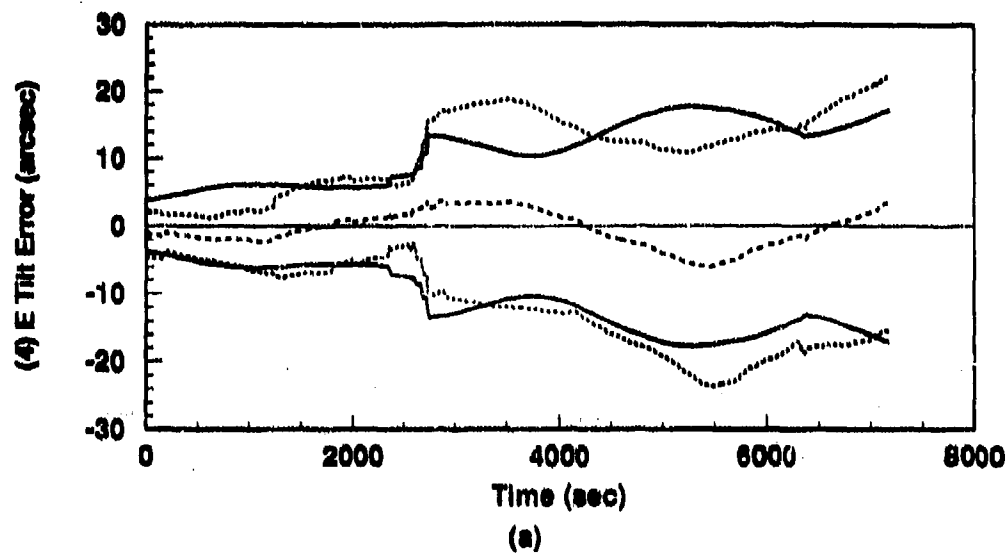


Figure D.8. Flight: 96-State Model (a) East Tilt and (b) North Tilt Error States. (Note: Initial conditions are those chosen by Litton.)

.....	Mean Error = $\bar{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

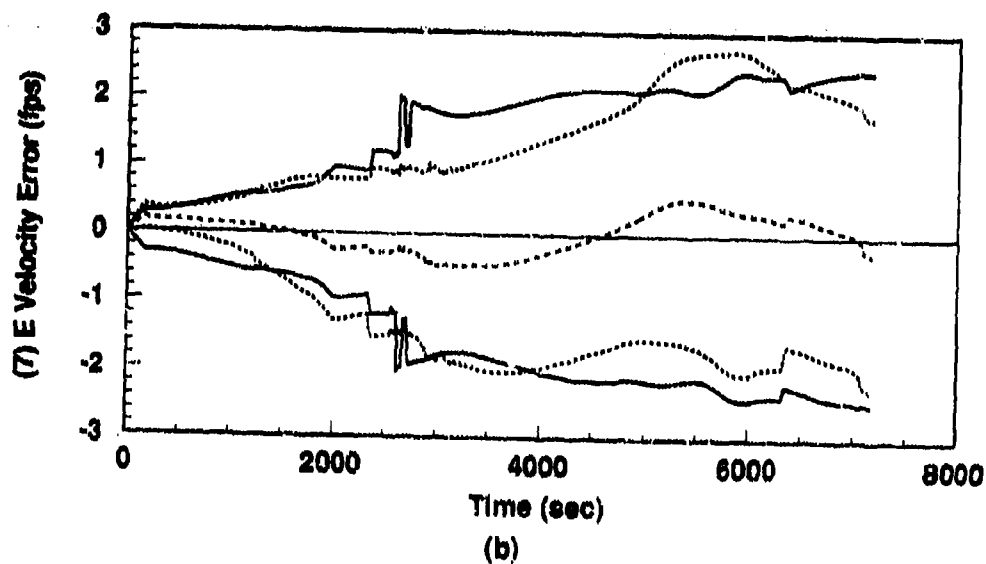
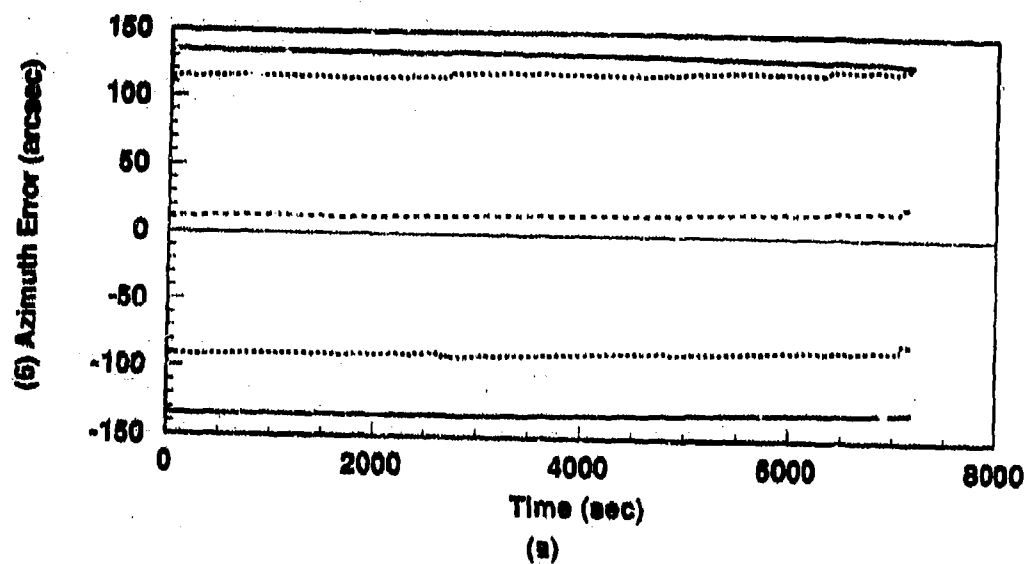


Figure D.9. Flight: 96-State Model (a) Azimuth and (b) East Velocity Error States.
(Note: Initial conditions are those chosen by Litton.)

---	Mean Error = $\bar{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

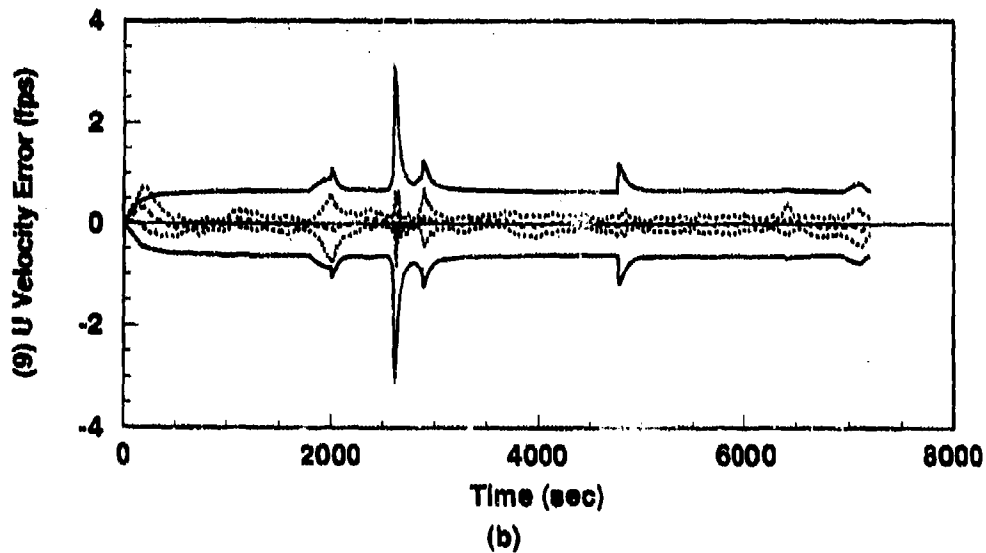
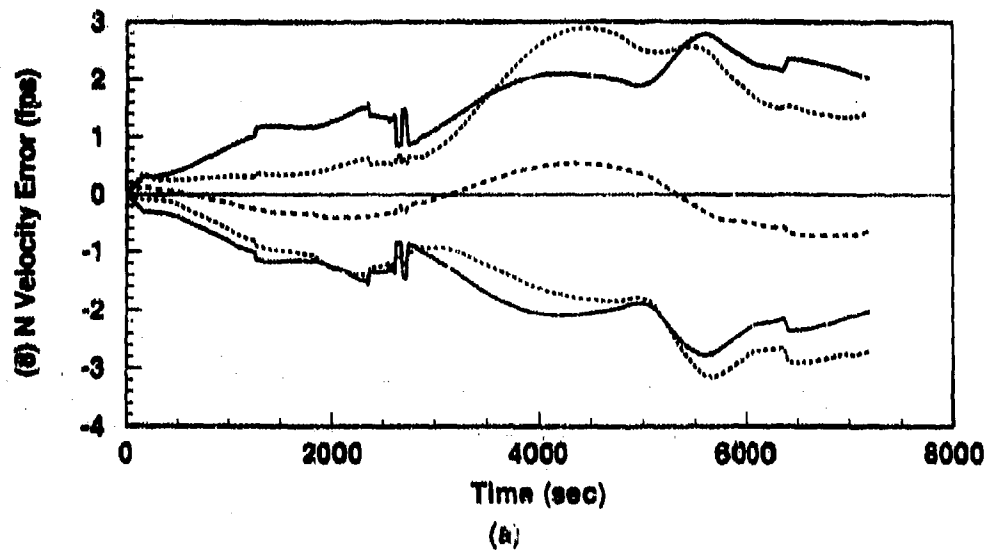


Figure D.10. Flight: 96-State Model (a) North Velocity and (b) Vertical Velocity Error States. (Note: Initial conditions are those chosen by Litton.)

----	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

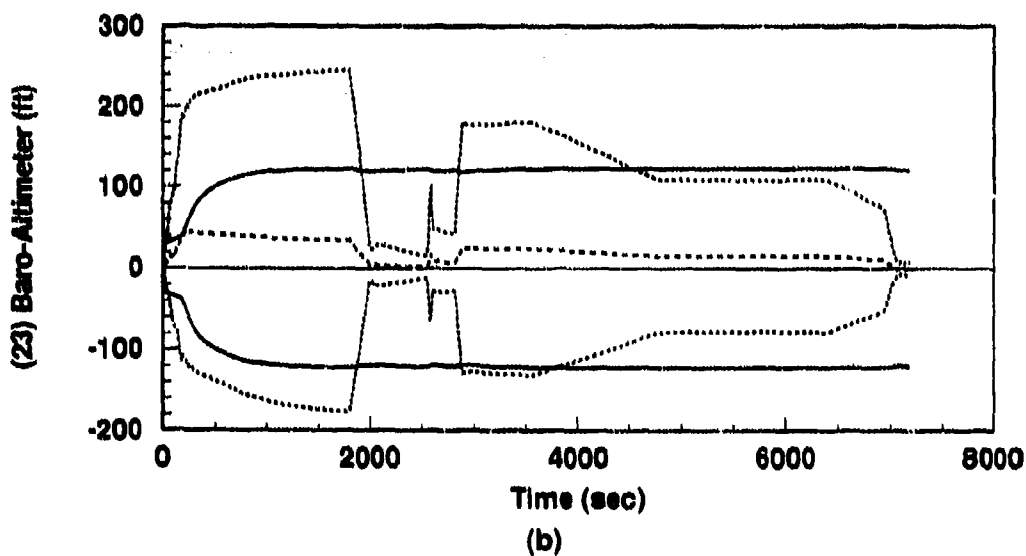
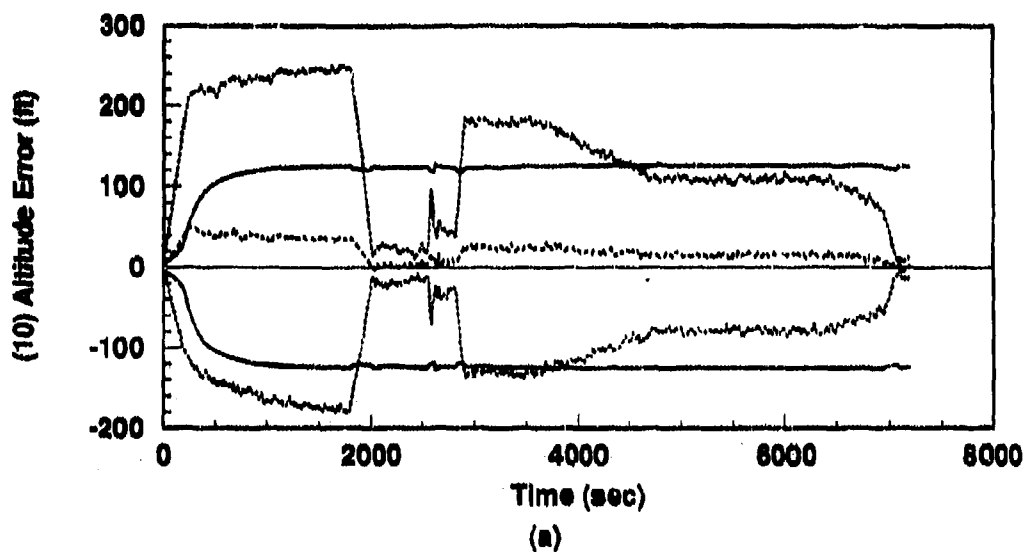


Figure D.11. Flight: 96-State Model (a) INS Altitude Error State and (b) Baro-Altimeter Total Error. (Note: Initial conditions are those chosen by Litton.)

----	Mean Error = $\bar{M}_r - (M_r)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

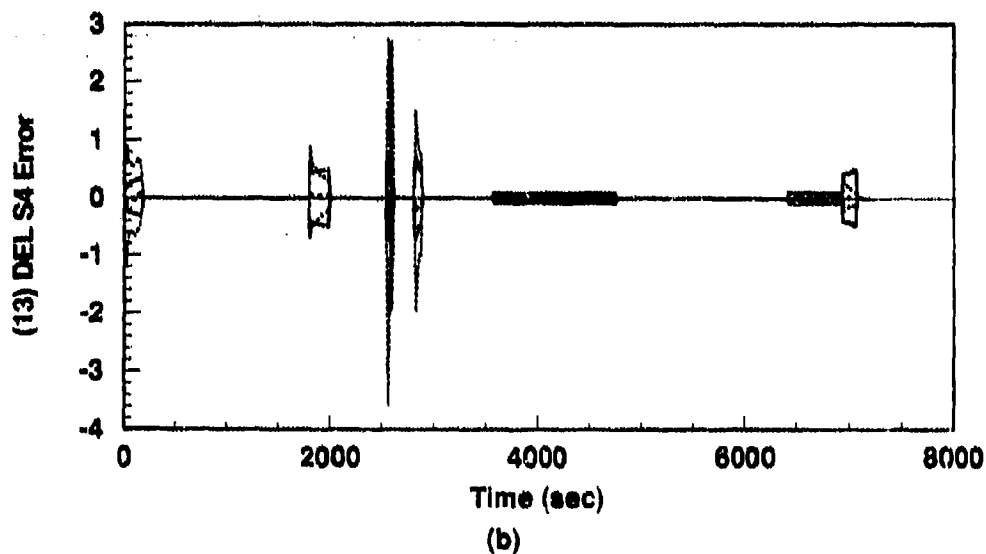
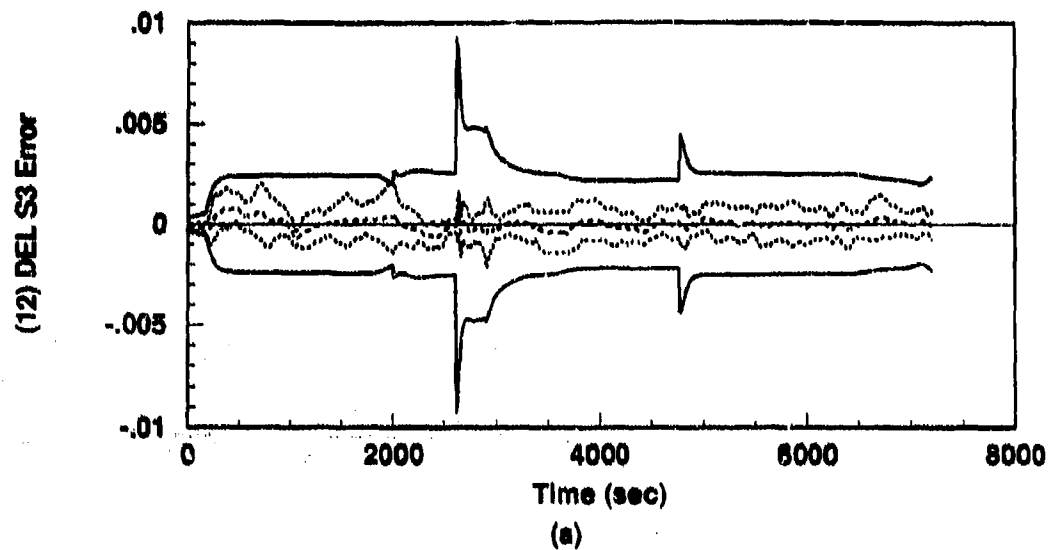


Figure D.12. Flight: 96-State Model (a) ΔS_3 and (b) ΔS_4 Vertical Channel Aiding Error States. (Note: Initial conditions are those chosen by Litton.)

----	Mean Error = $\bar{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
----	$0 \pm \sigma_{filter}$

Appendix E. 96-State INS Alignment and Flight Simulations: Part II

This appendix contains the plots from two sets of 10-run Monte Carlo simulations of the 96-state error model which incorporates the revised baro-altimeter model. It differs from Appendix D in that *initial conditions are chosen to reflect an alignment at Holloman AFB, NM*. This step is taken in order to establish a performance baseline using the INS with baro-aiding only. Subsequent tests with ground transponders (see Appendix F) and GPS (see Appendix G) are compared to results from simulations in this set.

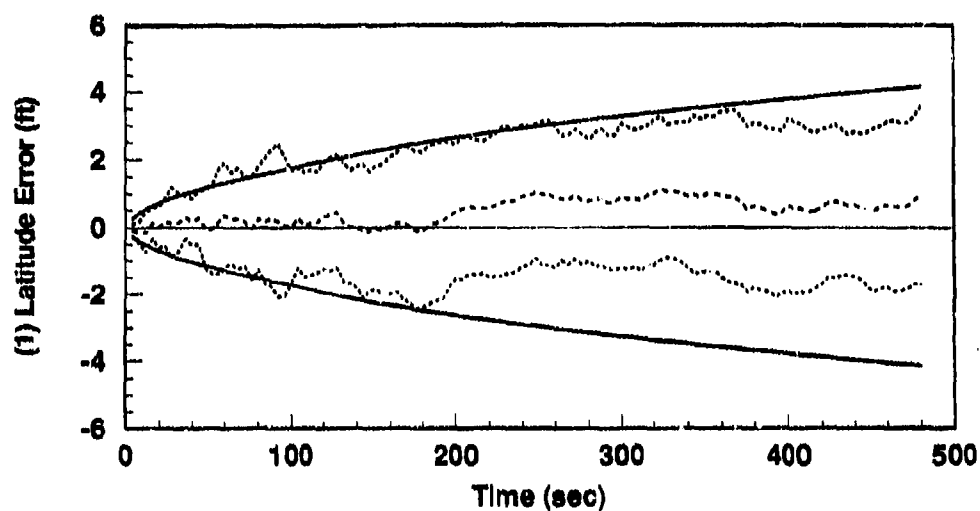
Variables plotted in this appendix are defined in exactly the same manner as those in Appendix C.

E.1 96-State Model: Alignment Using Holloman Initial Conditions

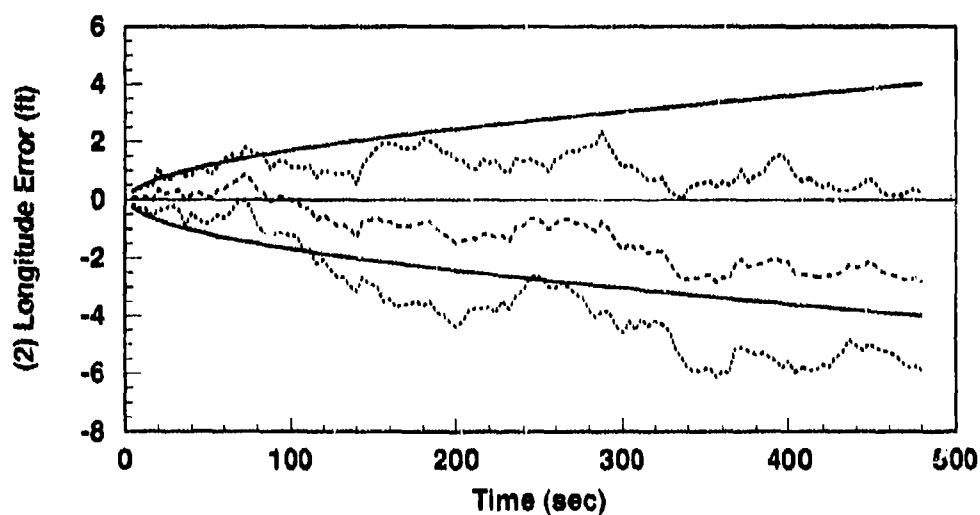
The plots in this section represent results of a 10-run Monte Carlo alignment simulation. In these alignments, INS aiding consists of velocity measurements plus baro-altimeter measurements.

Note that the plot for state 13, ΔS_1 , is zero for all time during all alignment simulations. This is a normal condition due to the variable's dependence on altitude rate. The state becomes non-zero during flight runs. All other plots are discussed in Chapter VI (Results).

The filter computed error estimates $[\sigma_{filter}]$ are also compared to similar plots contained in the Litton reference documentation [20]. Comparisons are qualitative only and are intended to demonstrate trends. Such comparisons are contained in Chapter VI (Results).



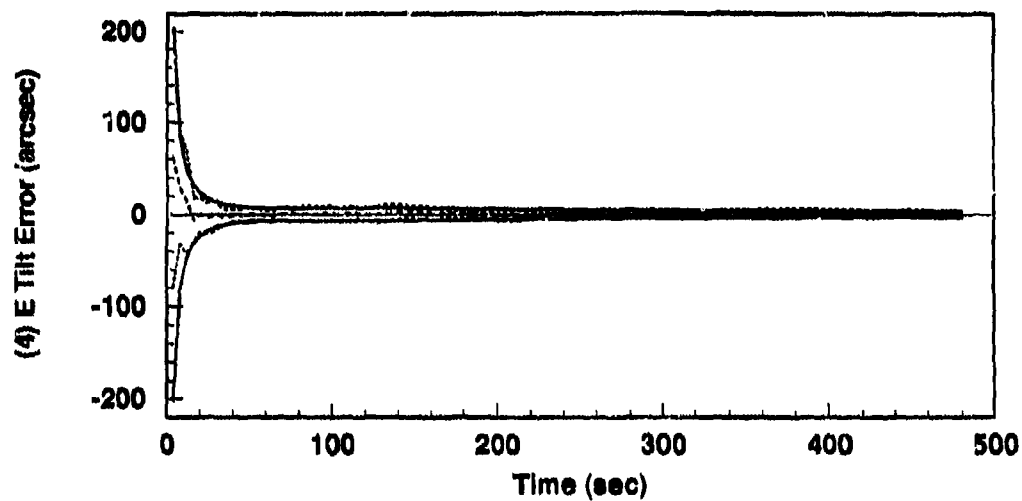
(a)



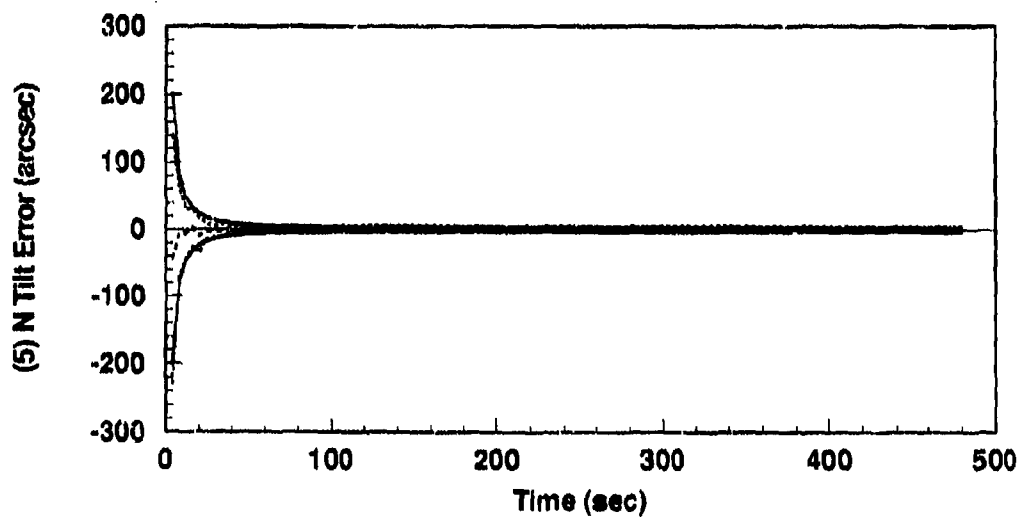
(b)

Figure E.1. Alignment: 96-State Model (a) Latitude and (b) Longitude Errors. (Note: Initial conditions are those for Holloman AFB.)

----	$Mean\ Error = \bar{M}_x - (M_x)_{true}$
.....	$Mean\ Error \pm \sigma_{true}$
-----	$0 \pm \sigma_{filter}$



(a)



(b)

Figure E.2. Alignment: 96-State Model (a) East Tilt and (b) North Tilt Error States.
(Note: Initial conditions are those for Holloman AFB.)

----	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

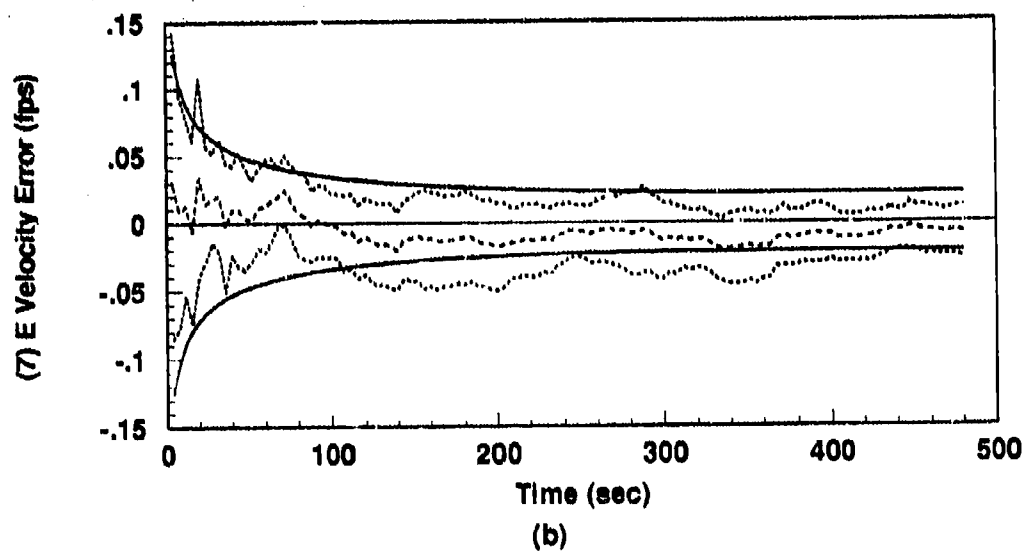
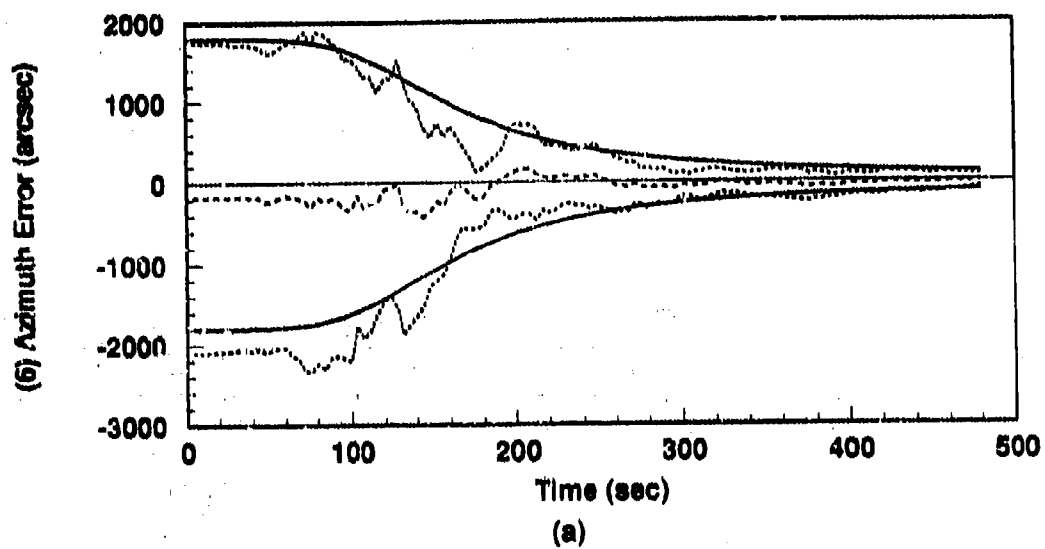
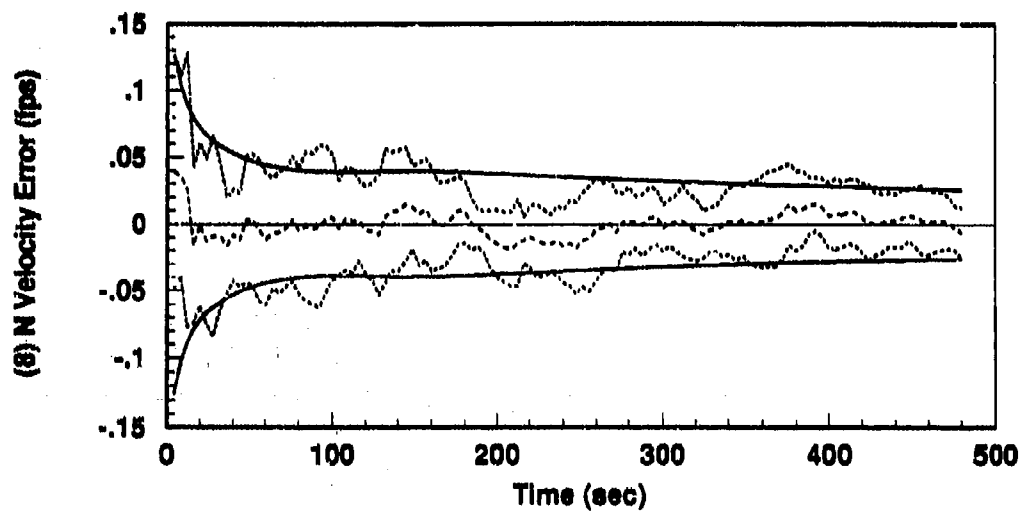
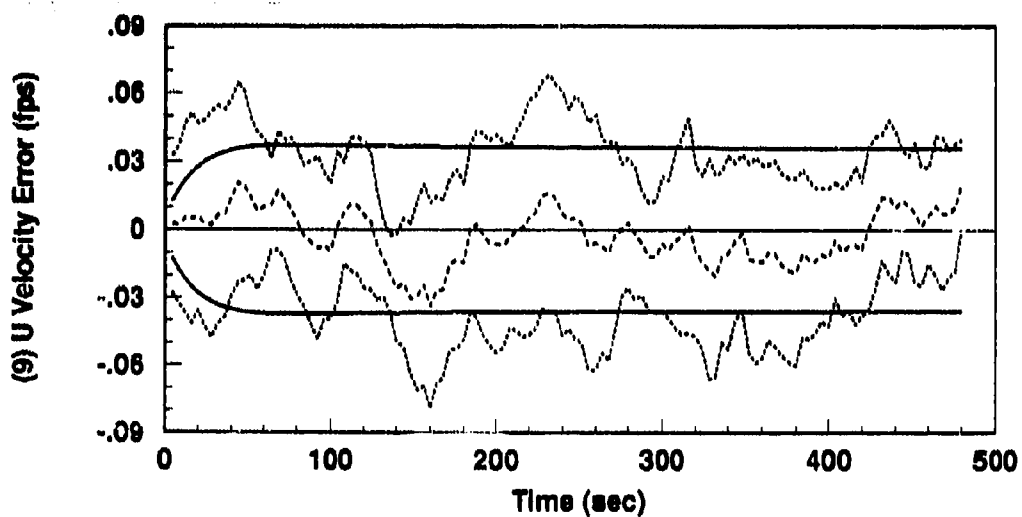


Figure E.3. Alignment: 96-State Model (a) Azimuth and (b) East Velocity Error States.
(Note: Initial conditions are those for Holloman AFB.)

----	$Mean\ Error = \hat{M}_x - (M_x)_{true}$
.....	$Mean\ Error \pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$



(a)



(b)

Figure E.4. Alignment: 96-State Model (a) North Velocity and (b) Vertical Velocity Error States. (Note: Initial conditions are those for Holloman AFB.)

----	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

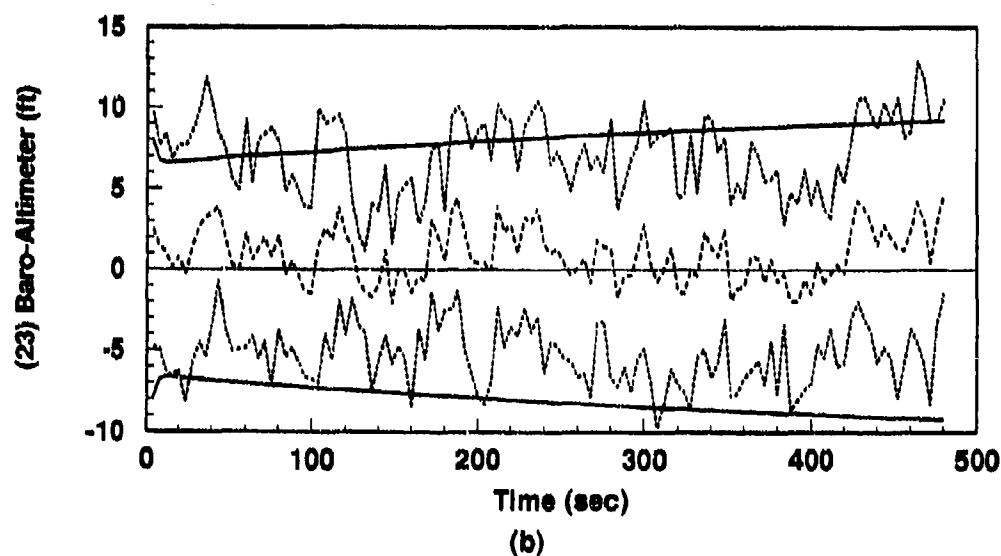
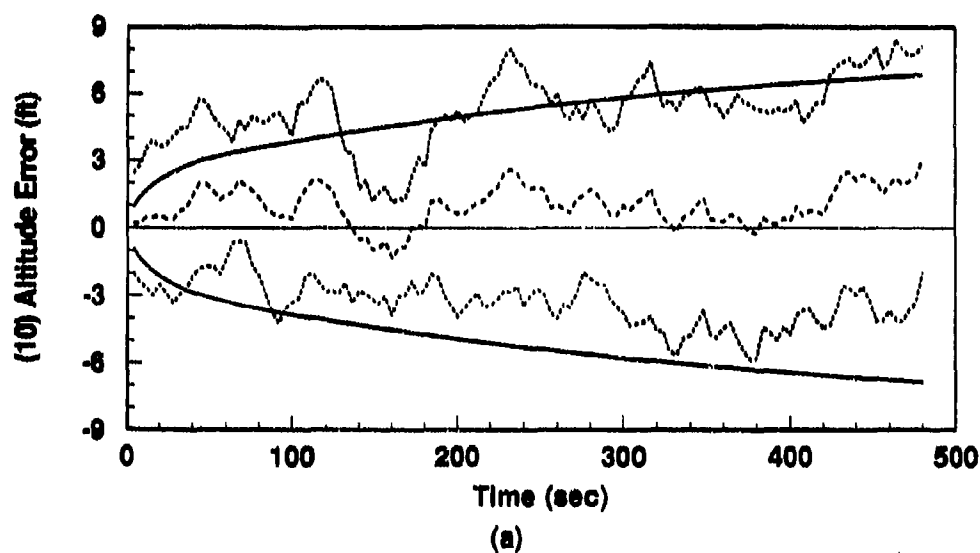


Figure E.5. Alignment: 96-State Model (a) INS Altitude Error State and (b) Baro-Altimeter Total Error. (Note: Initial conditions are those for Holloman AFB.)

----	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

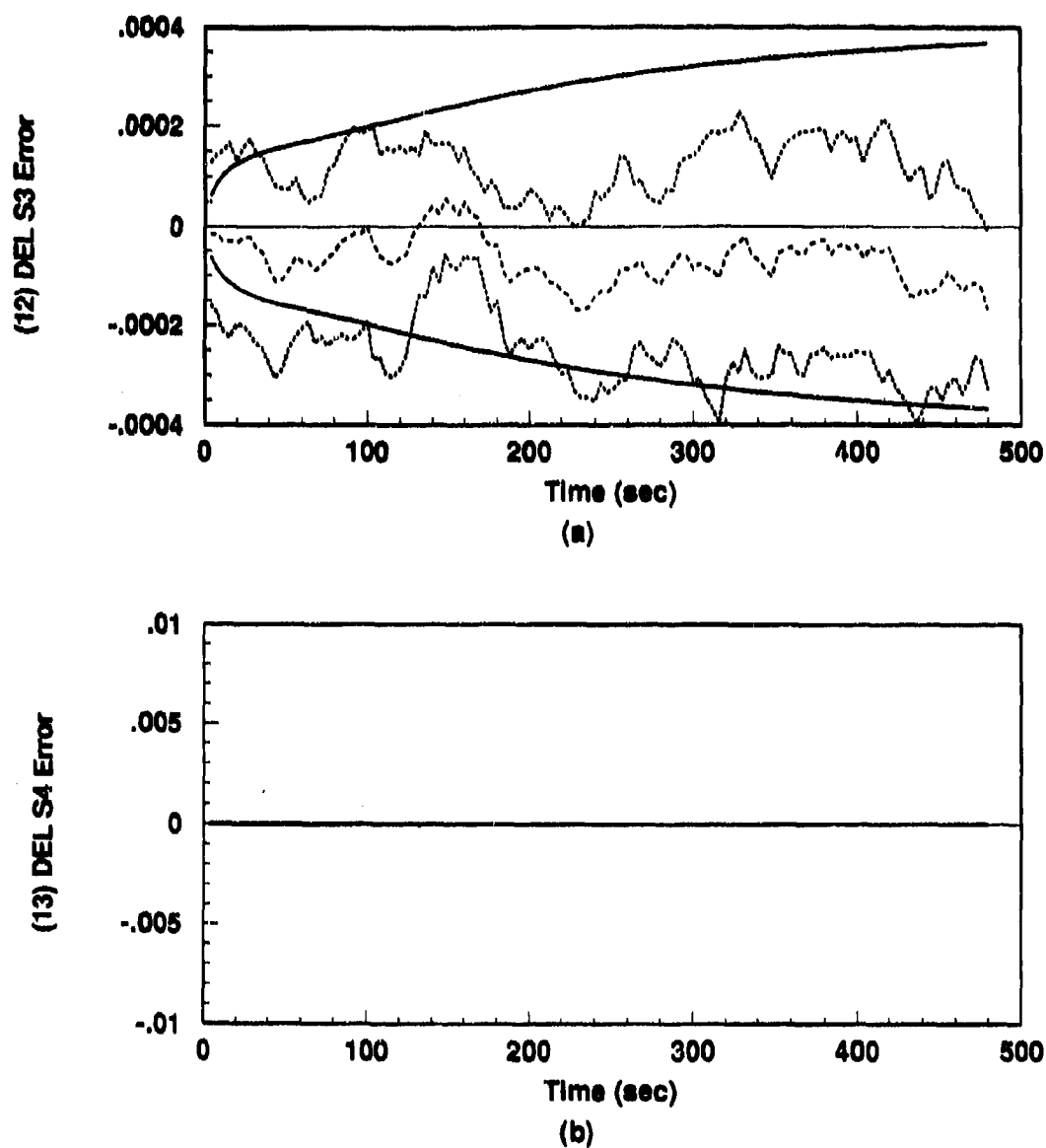


Figure E.6. Alignment: 96-State Model (a) ΔS_3 and (b) ΔS_4 Vertical Channel Aiding Error States. (Note: Initial conditions are those for Holloman AFB.)

.....	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

E.2 96-State Model: Fighter Flight Using Holloman Initial Conditions

The plots in this section represent results of a 10-run flight simulation in which a flight profile (as described in Chapter III) is used to characterize the LN-93 performance for a fighter mission originating and terminating at Holloman AFB, NM. For this group of runs, only baro-altimeter aiding is used.

The purpose of this set of runs is to establish that the software function for a flight using Holloman coordinates results in performance which is comparable to that achieved in the Litton flight runs reported in Appendix C.

The filter computed error estimates [σ_{filter}] are also compared to similar plots contained in the Litton reference documentation [20]. Comparisons are qualitative only and are intended to demonstrate trends. Such comparisons are contained in Chapter VI (Results).

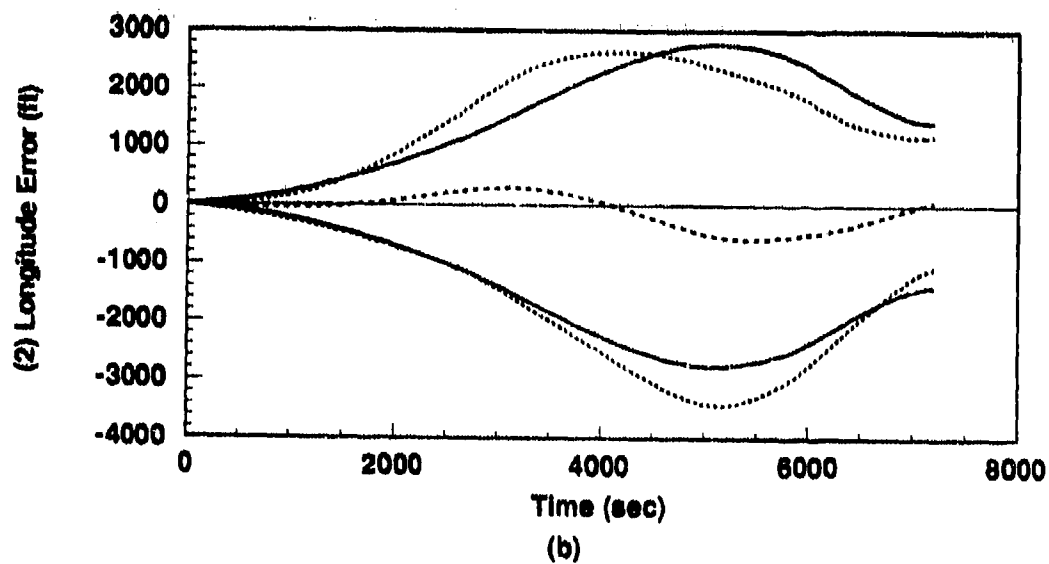
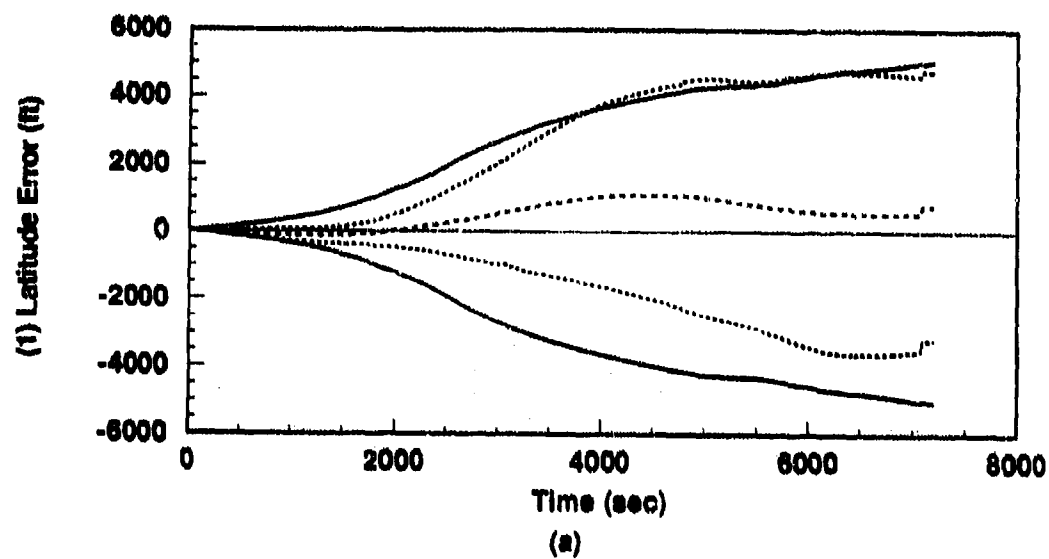


Figure E.7. Flight: 96-State Model (a) Latitude and (b) Longitude Error States. (Note: Initial conditions are those for Holloman AFB.)

....	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

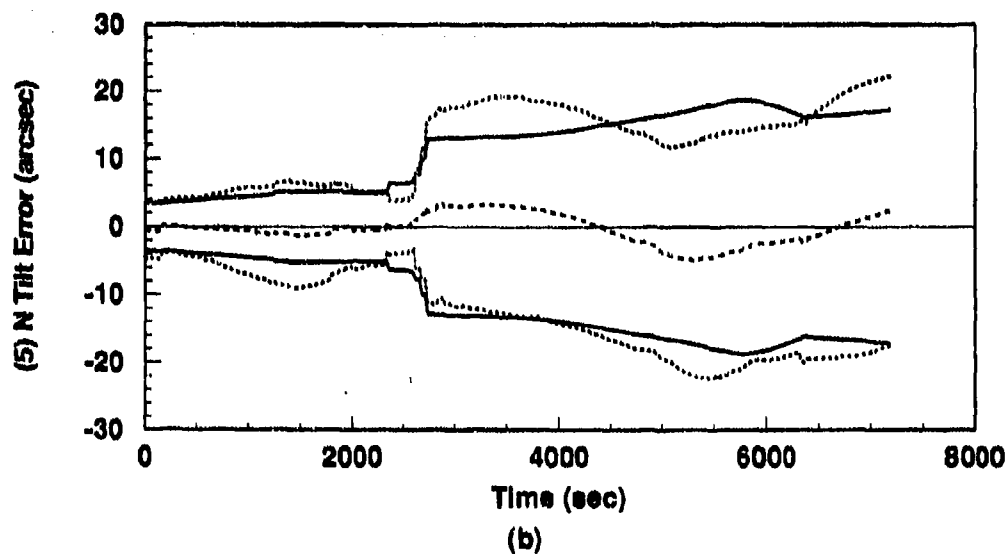
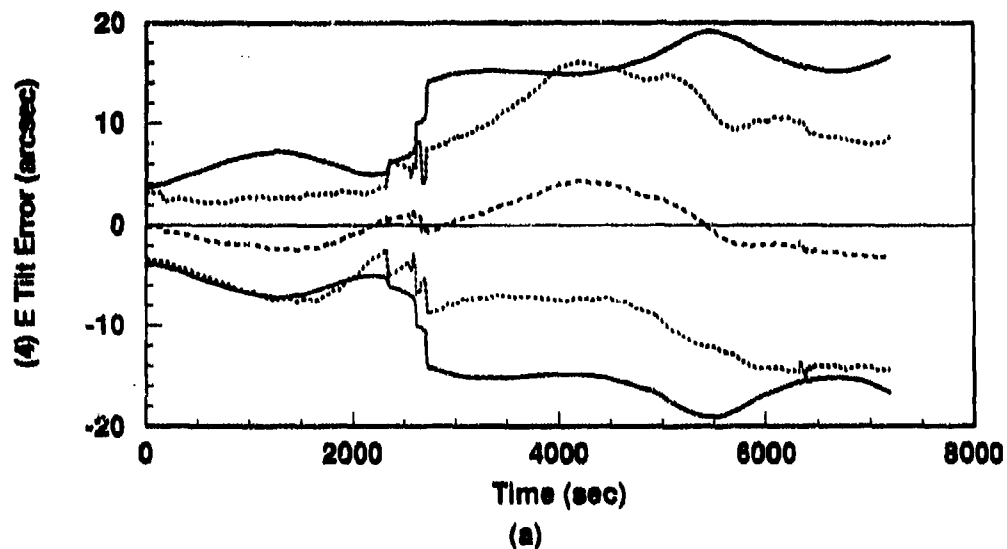


Figure E.8. Flight: 96-State Model (a) East Tilt and (b) North Tilt Error States. (Note: Initial conditions are those for Holloman AFB.)

----	Mean Error = $\bar{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

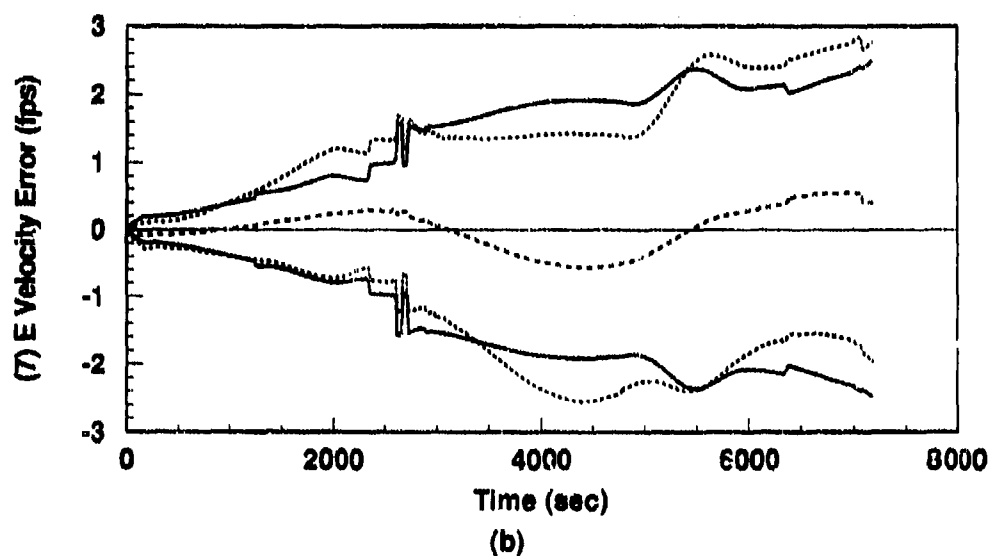
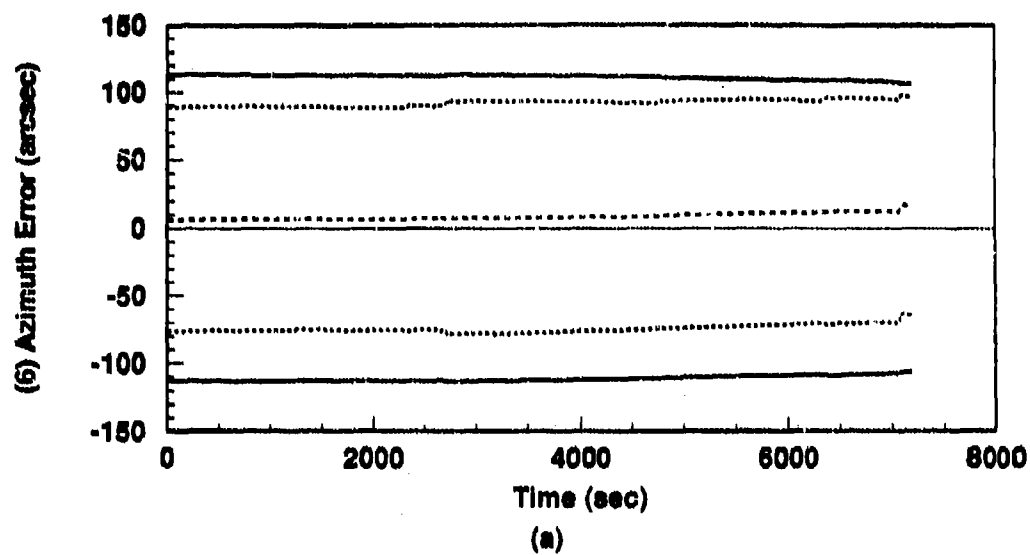


Figure E.9. Flight: 96-State Model (a) Azimuth and (b) East Velocity Error States.
(Note: Initial conditions are those for Holloman AFB.)

----	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

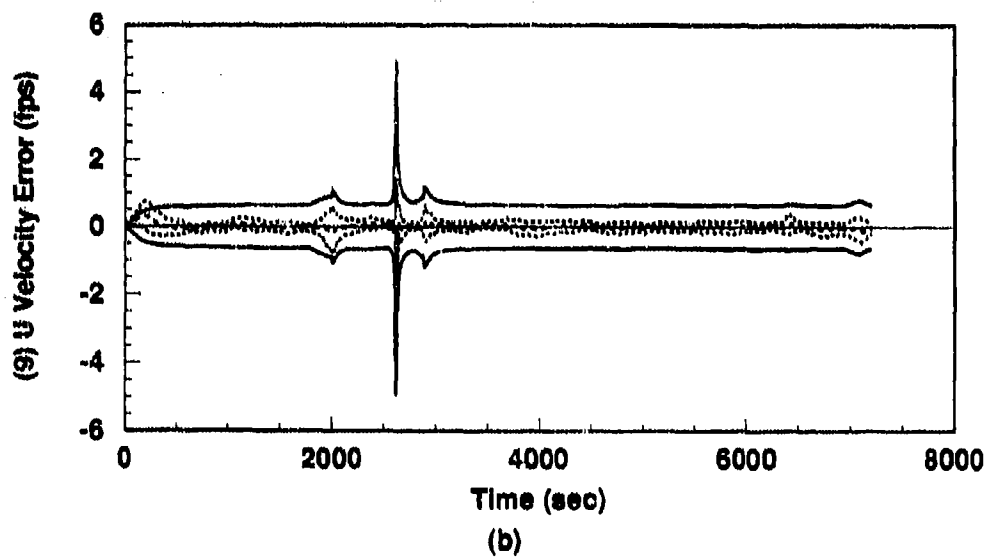
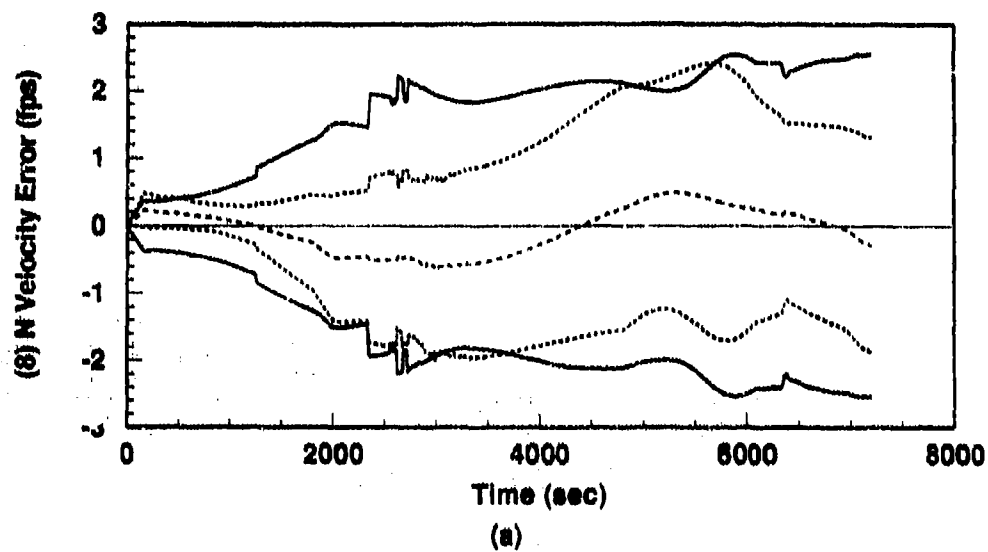


Figure E.10. Flight: 96--State Model (a) North Velocity and (b) Vertical Velocity Error States. (Note: Initial conditions are those for Holloman AFB.)

----	$\text{Mean Error} = \bar{M}_x - (M_x)_{true}$
.....	$\text{Mean Error} \pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

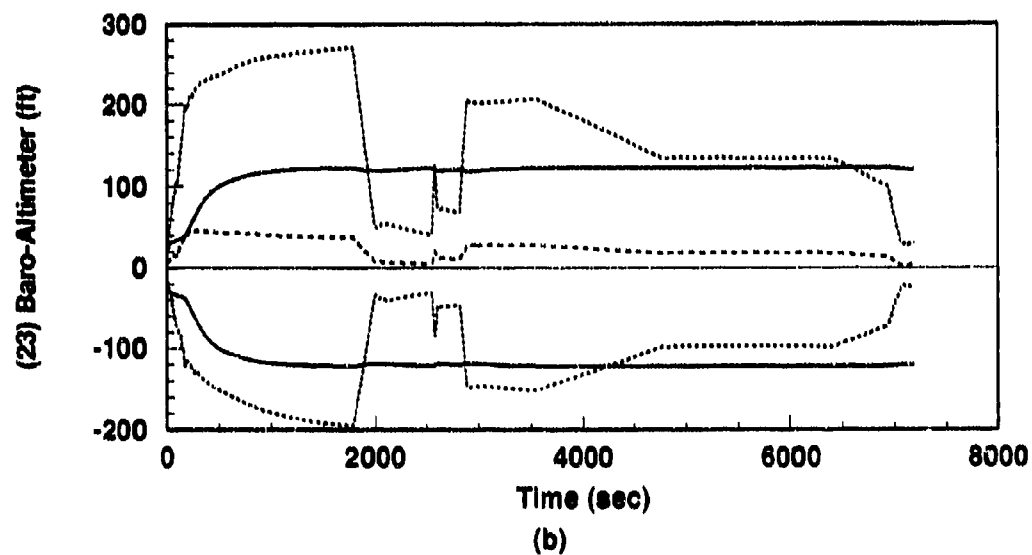
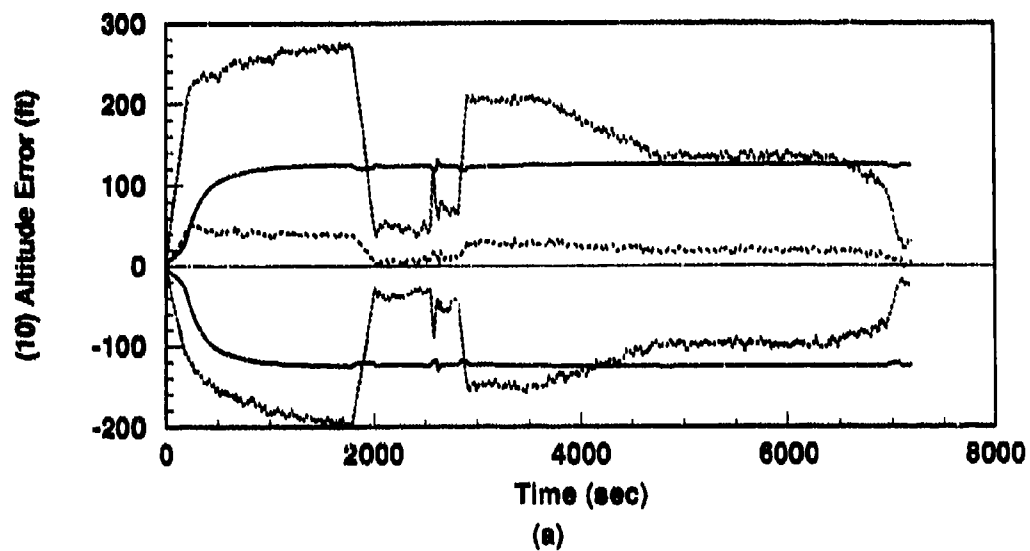


Figure E.11. Flight: 96-State Model (a) INS Altitude Error State and (b) Baro-Altimeter Total Error. (Note: Initial conditions are those for Holloman AFB.)

----	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

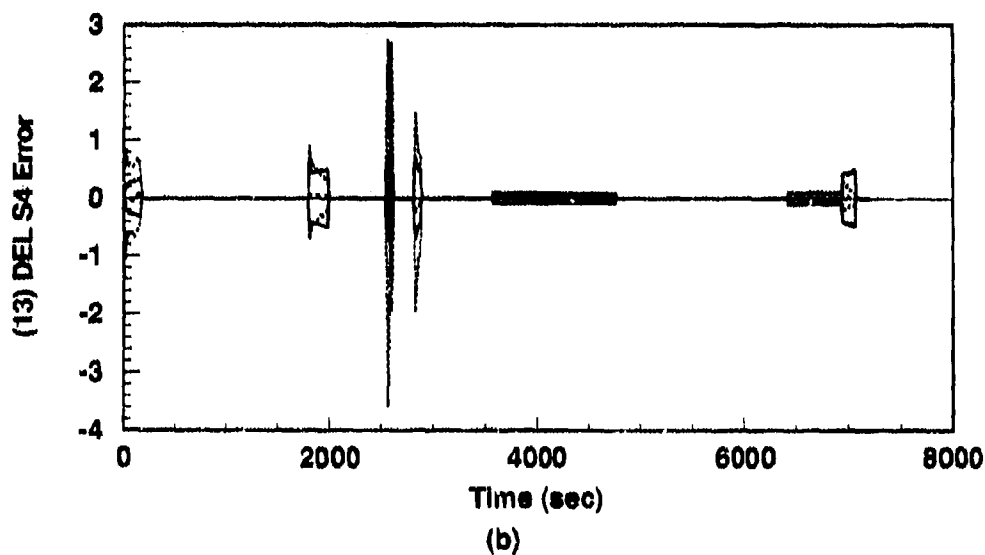
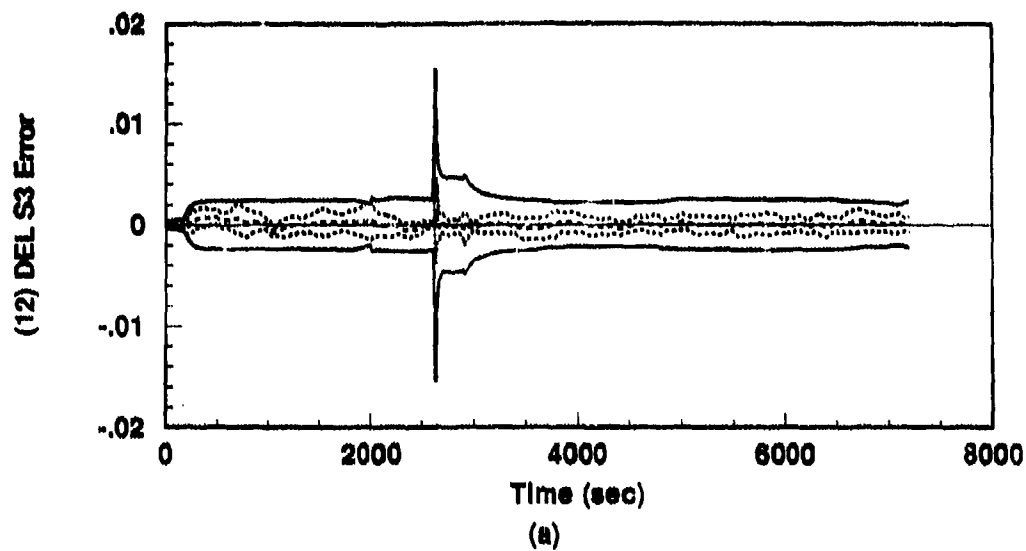


Figure E.12. Flight: 96-State Model (a) ΔS_3 and (b) ΔS_4 Vertical Channel Aiding Error States. (Note: Initial conditions are those for Holloman AFB.)

----	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
-----	$0 \pm \sigma_{filter}$

Appendix F. 72-State Reduced INS Model Validation

This appendix contains the plots from a 10-run Monte Carlo alignment simulation of a 72-state error model which incorporates a *reduced* INS model and a revised baro-altimeter model. The 72-state *truth model* one sigma values obtained during this simulation are compared directly to the 96-state *truth model* one sigma values obtained during the alignment runs presented in Appendix E. The purpose of this comparison is to ensure that the 72 state model adequately represents the 96 state model.

Prior to the error model reduction discussed above, truth models consisting of first 122 and finally 152 states had been programmed into MSOFE. The 122-state version (which integrated the 96-state INS and the 26-state RRS models) was successfully compiled and run. Subsequently, the GPS error model was integrated into the overall error model, bringing the truth and Kalman filter error model dimensions to 152 states each. At that point, hardware/compiler limitations prevented compiling MSOFE to run on the accelerator board. Inadequate random access memory (RAM) is believed to be the cause of difficulty. In order to continue to use the accelerator board, the size of the problem had to be reduced.

As a result, several of the less important states contained in the LN-93 error model are eliminated. Certain states are chosen for elimination based on previous research [11, 19], and validation of the selection is accomplished in a series of Monte Carlo runs which culminate in the plots presented in this appendix.

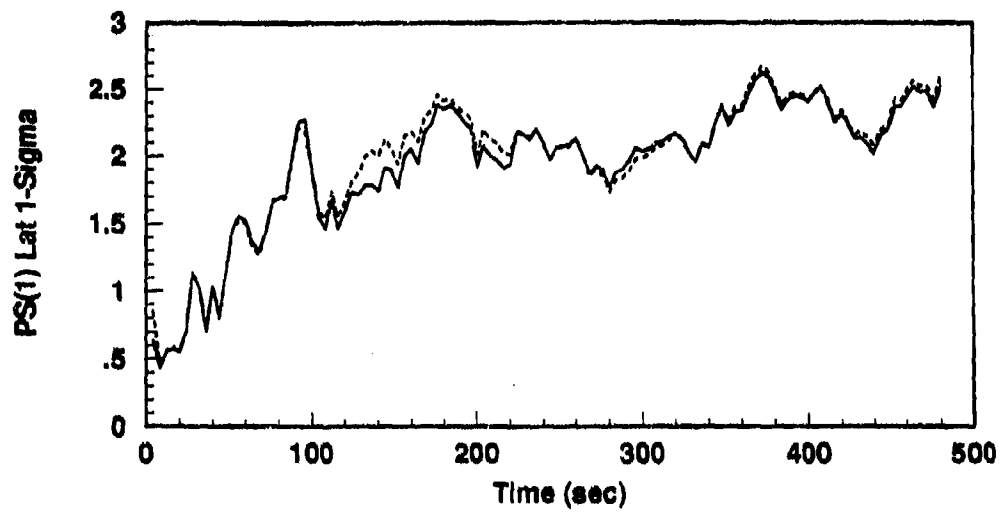
Although 24 states have been eliminated from the INS error model, the plots contained in this appendix demonstrate that no significant loss of fidelity results. In all important states (i.e., position, platform tilts, and velocities), the reduced *system model* accurately portrays the error behavior of the full 96-state truth model.

Variables plotted in this appendix are simply the standard deviation time histories for several *truth model* states of interest.

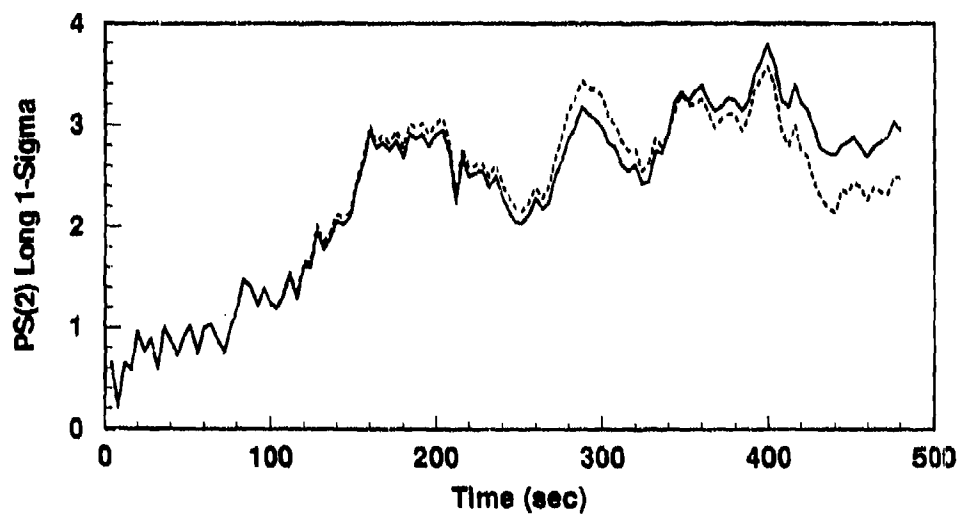
F.1 72-State Model: Alignment Using Holloman Initial Conditions

The plots in this section represent results of a 10-run Monte Carlo alignment simulation of the reduced INS error model. In these alignments, INS aiding consists of velocity measurements plus baro-altimeter measurements only.

The filter computed error estimates $[\sigma_{filter}]$ are also compared to similar plots contained in the Litton reference documentation [20]. Comparisons are qualitative only and are intended to demonstrate trends. Such comparisons are contained in Chapter VI (Results).



(a)



(b)

Figure F.1. Alignment: 96-State vs. 72-State Model (a) Latitude and (b) Longitude Error States. (Note: Initial conditions are those for Holloman AFB.)

—	96-State Truth Model Error Behavior
- - -	72-State Truth Model Error Behavior

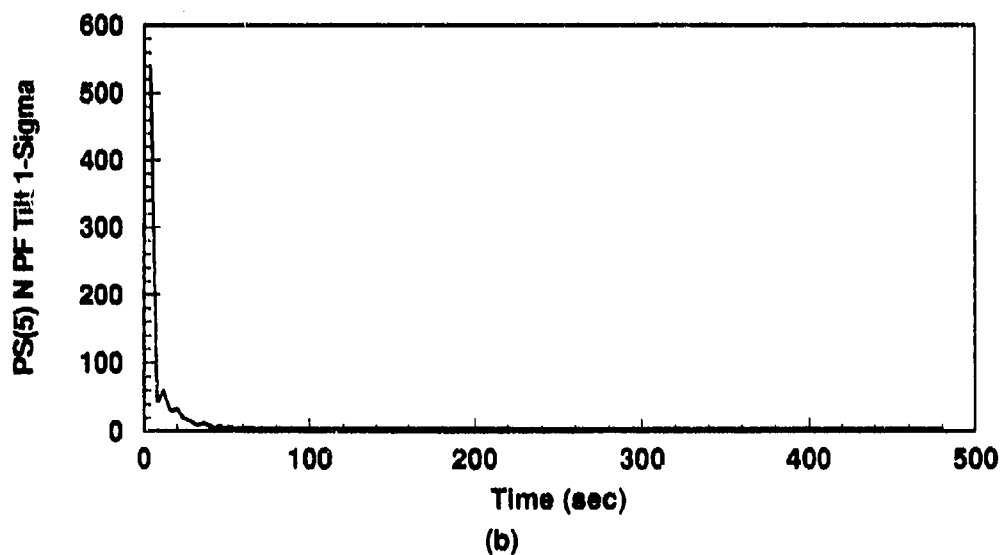
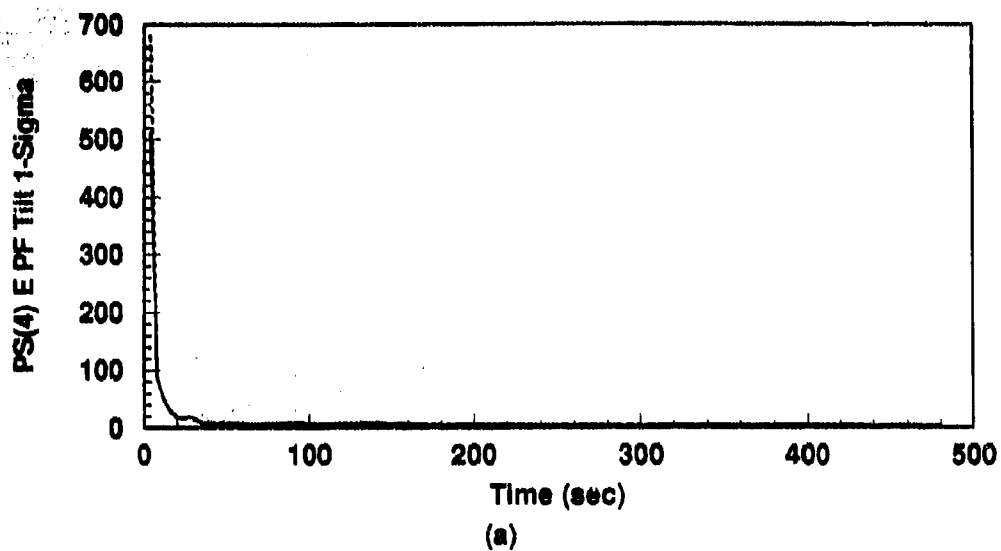


Figure F.2. Alignment: 96-State vs. 72-State Model (a) East Tilt and (b) North Tilt Error States. (Note: Initial conditions are those for Holloman AFB.)

—	96-State Truth Model Error Behavior
- - -	72-State Truth Model Error Behavior

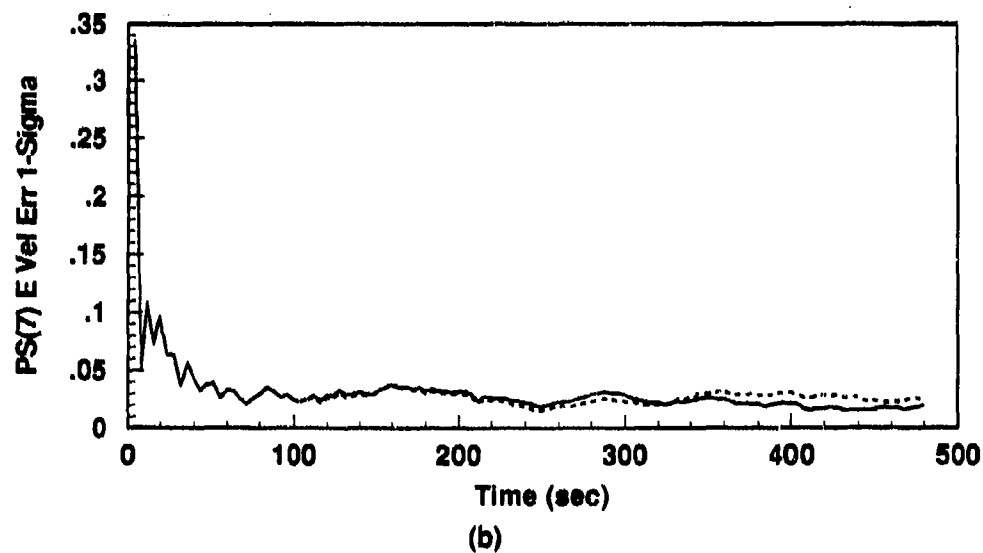
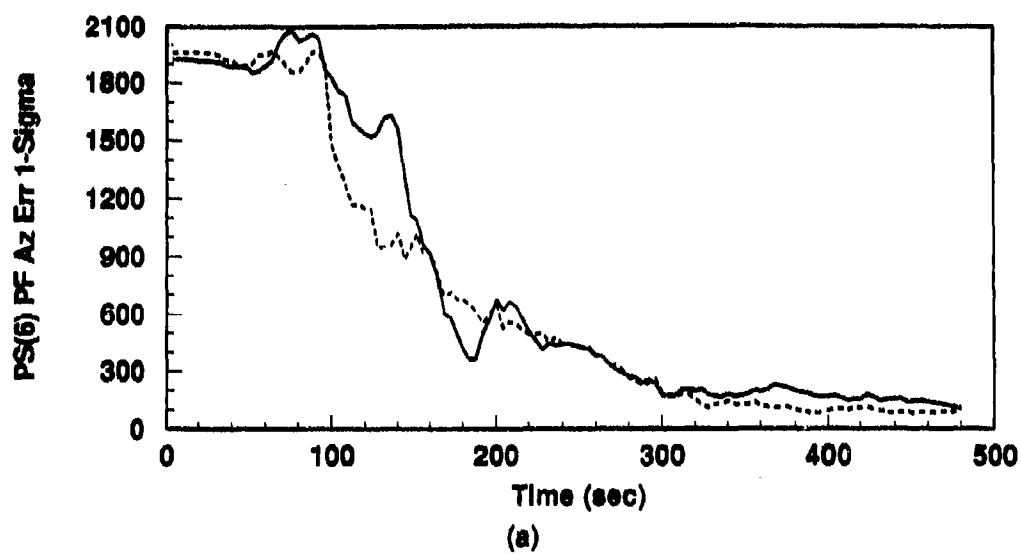


Figure F.3. Alignment: 96-State vs. 72-State Model (a) Azimuth and (b) East Velocity Error States. (Note: Initial conditions are those for Holloman AFB.)

—	96-State Truth Model Error Behavior
----	72-State Truth Model Error Behavior

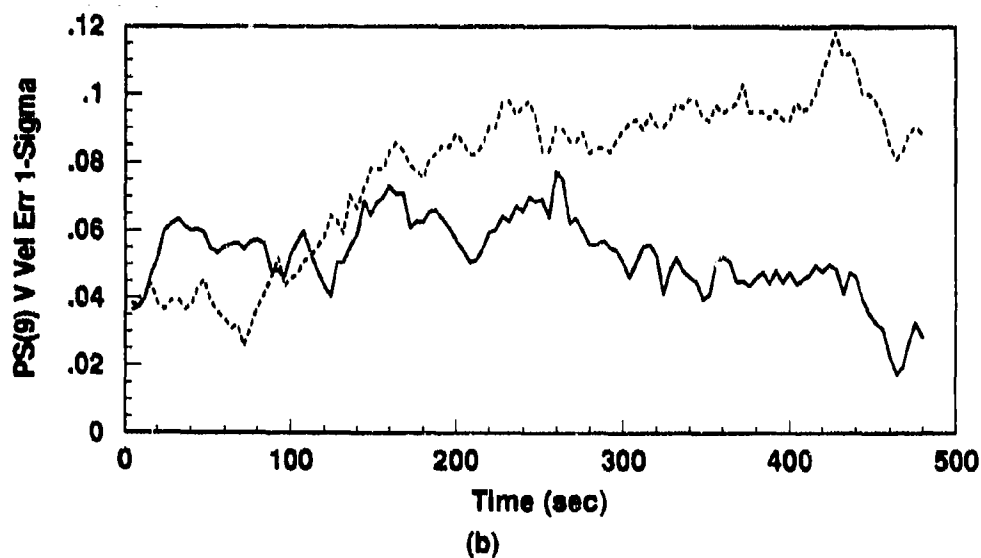
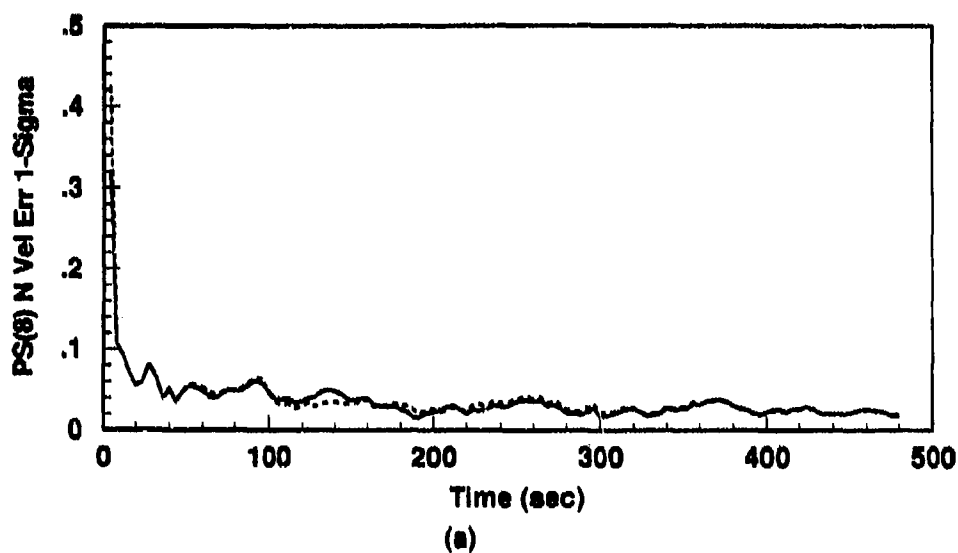


Figure F.4. Alignment: 96-State vs. 72-State Model (a) North Velocity and (b) Vertical Velocity Error States. (Note: Initial conditions are those for Holloman AFB.)

—	96-State Truth Model Error Behavior
...	72-State Truth Model Error Behavior

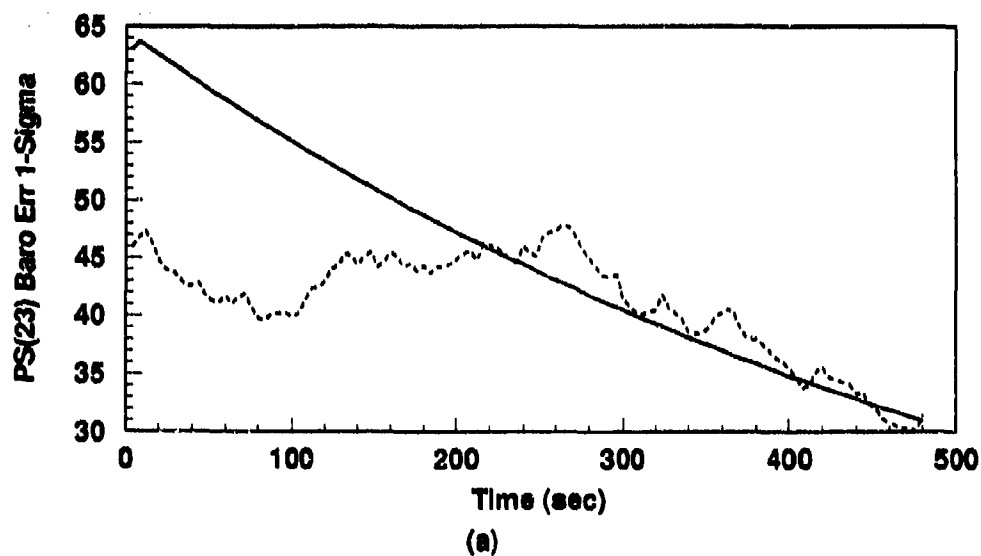
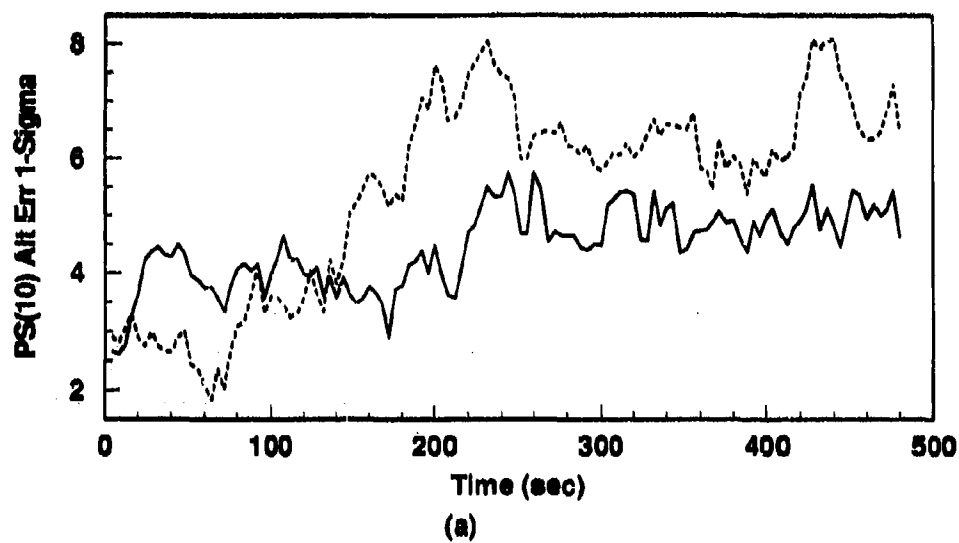


Figure F.5. Alignment: 96-State vs. 72-State Model (a) INS Altitude Error State and (b) Baro-Altimeter Total Error. (Note: Initial conditions are those for Holloman AFB.)

—	96-State Truth Model Error Behavior
- - -	72-State Truth Model Error Behavior

Appendix G. 98-State INS-RRS Alignment and Flight Simulations

This appendix contains the plots one 10-run Monte Carlo alignment and one 5-run Monte Carlo flight simulation of the 98-state error model which incorporates the *truncated* LN-93 INS, the revised baro-altimeter model, and RRS. Initial conditions are again chosen to reflect an alignment at Holloman AFB, NM. This step is important because *actual* surveyed positions for the RRS transponders are used in this study (the RRS transponders are physically located on or near Holloman AFB). These results (using ground transponders) are compared to results from simulations in Appendix F and Appendix H.

Variables plotted in this appendix are defined exactly in the same manner as those in Appendix C; all statistical quantities are calculated in the manner described at the beginning of Appendix C.

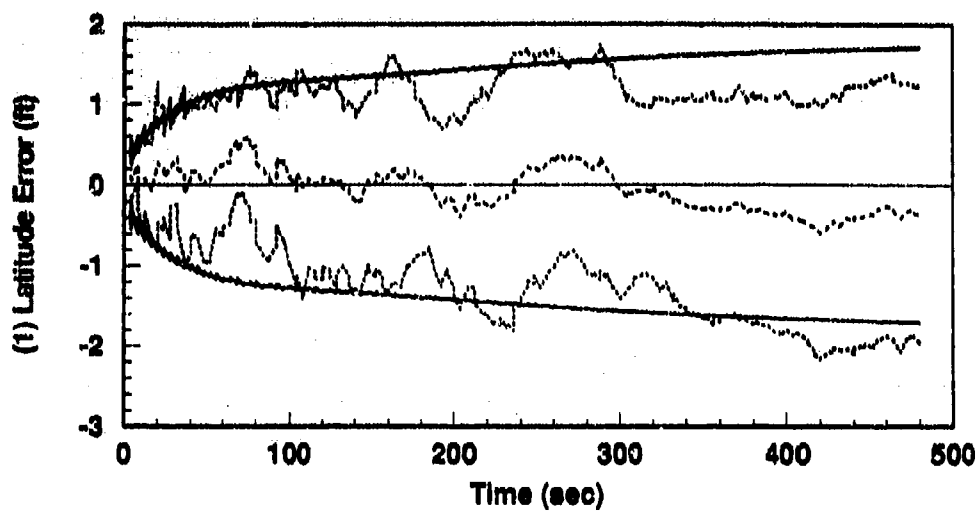
G.1 98-State Model: Alignment Using Holloman Initial Conditions

The plots in this section represent results of a 10-run Monte Carlo alignment simulation. In these alignments, INS aiding consists of velocity measurements, baro-altimeter measurements, and RRS transponder range measurements.

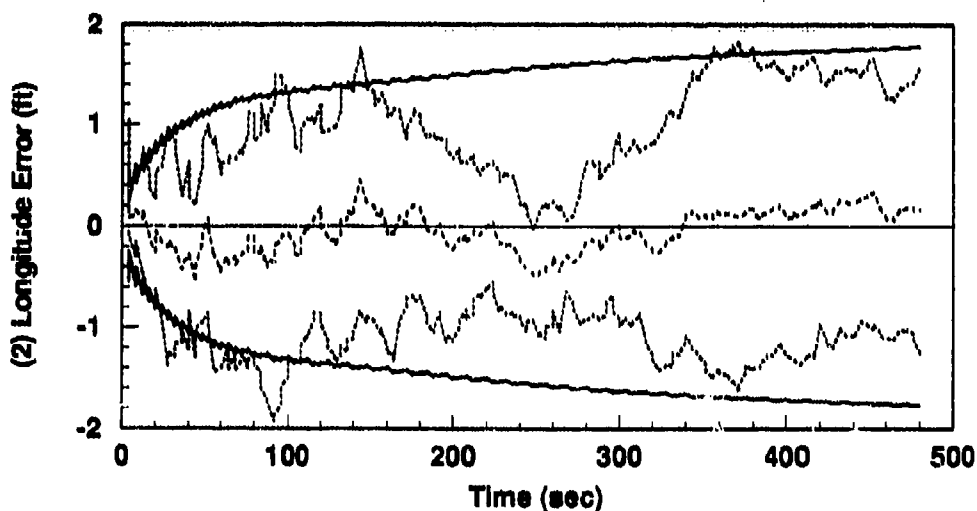
The purpose of this set of runs is to establish that the software function for the INS-RRS combination is comparable to (or better than) that achieved in the Litton flight runs reported in Appendices C through E.

Note that the plot for state 13, ΔS_4 , is zero for all time during all alignment simulations. This is a normal condition due to the variable's dependence on altitude rate. The state becomes non-zero during flight runs. All other plots are discussed in Chapter VI (Results).

The filter computed error estimates $[\sigma_{filter}]$ are also compared to similar plots contained in the Litton reference documentation [20]. Comparisons are qualitative only and are intended to demonstrate trends. Such comparisons are contained in Chapter VI (Results).



(a)



(b)

Figure G.1. Alignment: 98-State Model (a) Latitude and (b) Longitude Errors. (Note: Initial conditions are those for Holloman AFB.)

.....	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
———	$0 \pm \sigma_{filter}$

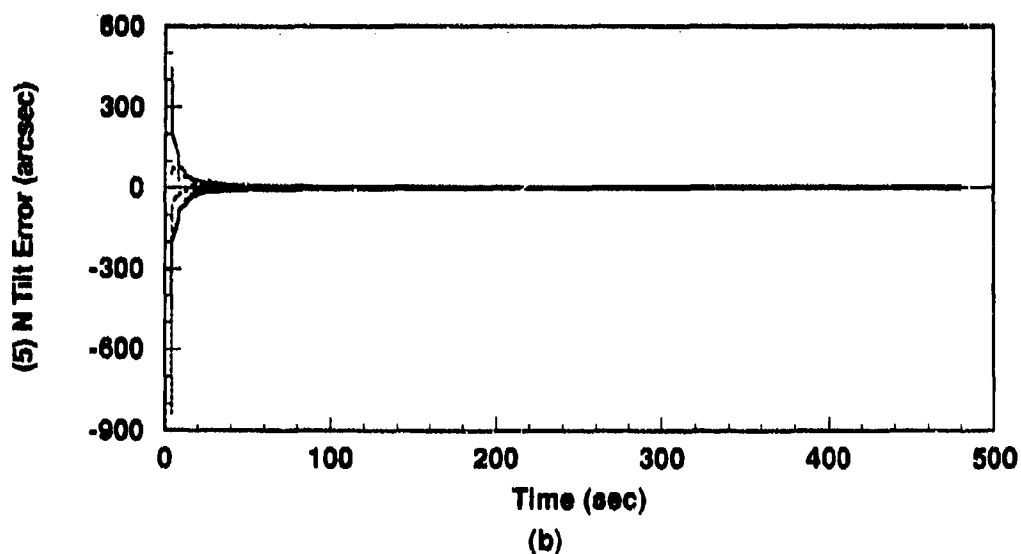
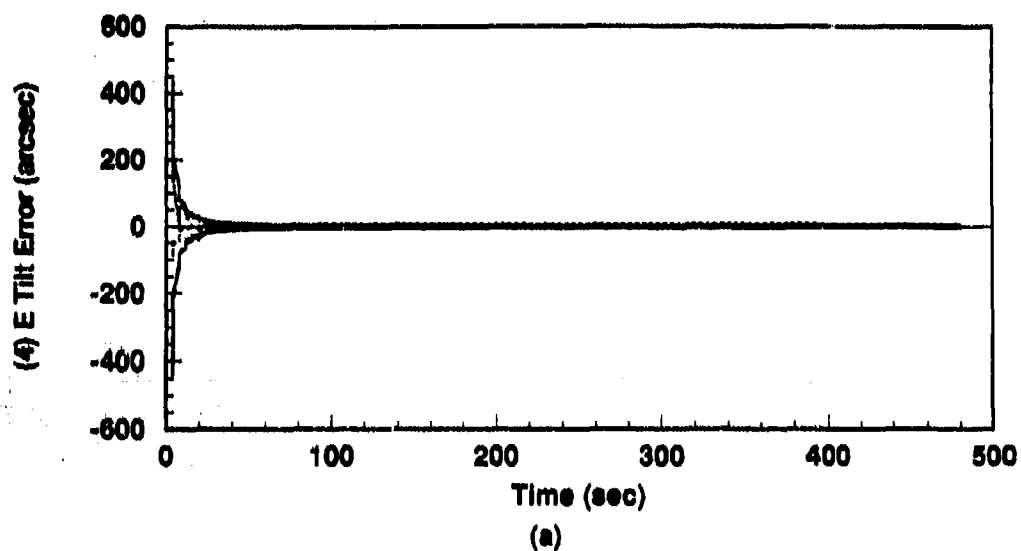


Figure G.2. Alignment: 98-State Model (a) East Tilt and (b) North Tilt Error States.
(Note: Initial conditions are those for Holloman AFB.)

----	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{trim}$
—	$0 \pm \sigma_{filter}$

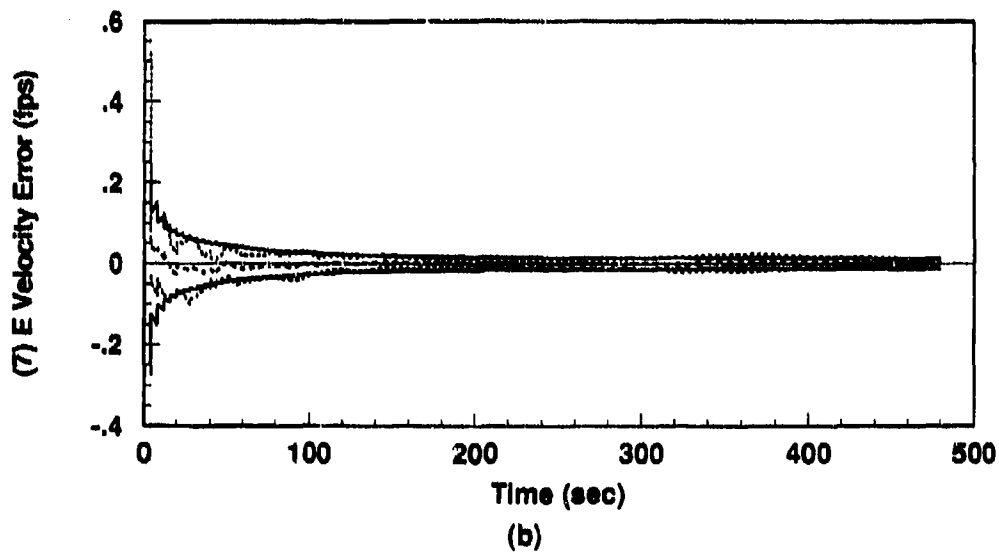
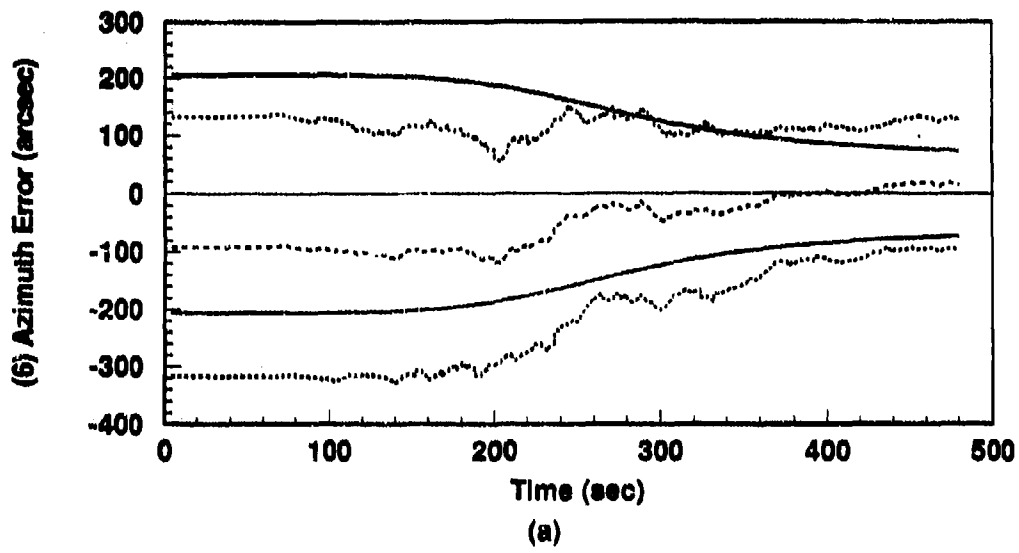


Figure G.3. Alignment: 98-State Model (a) Azimuth and (b) East Velocity Error States.
(Note: Initial conditions are those for Holloman AFB.)

----	Mean Error = $\bar{M}_r - (M_r)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

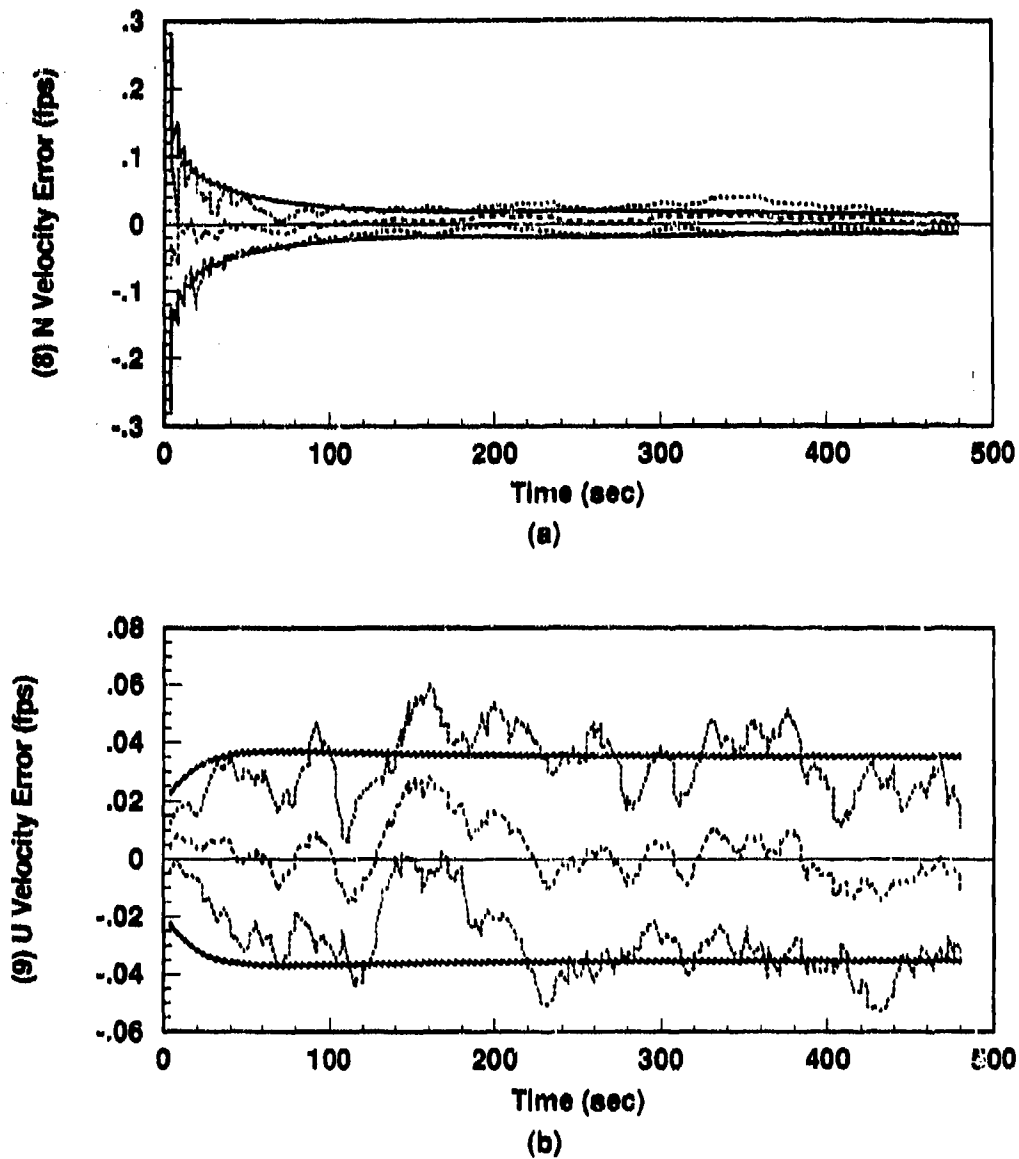


Figure G.4. Alignment: 98-State Model (a) North Velocity and (b) Vertical Velocity Error States. (Note: Initial conditions are those for Holloman AFB.)

----	Mean Error = $\bar{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
-----	$0 \pm \sigma_{filter}$

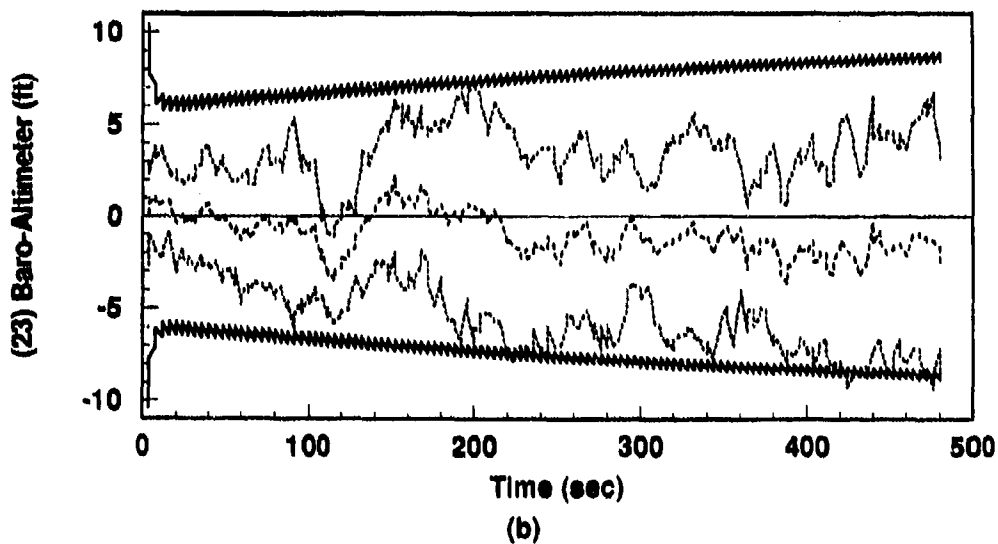
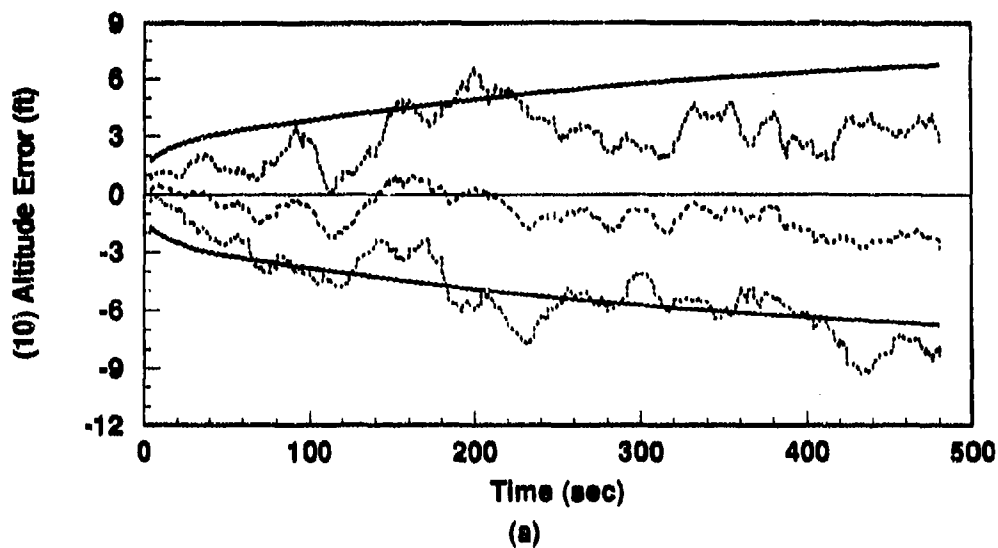


Figure G.5. Alignment: 98-State Model (a) INS Altitude Error State and (b) Baro-Altimeter Total Error. (Note: Initial conditions are those for Holloman AFB.)

----	Mean Error = $\bar{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

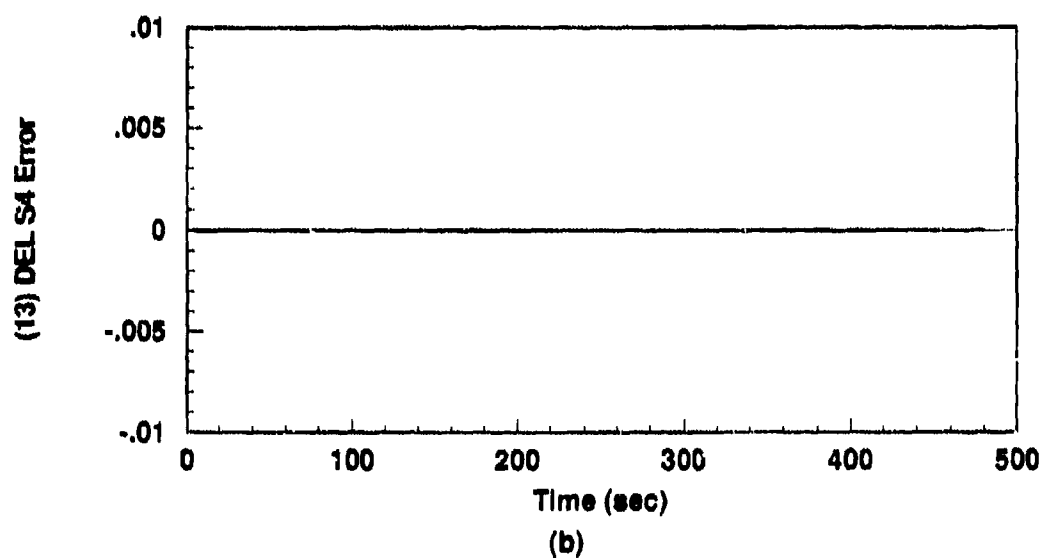
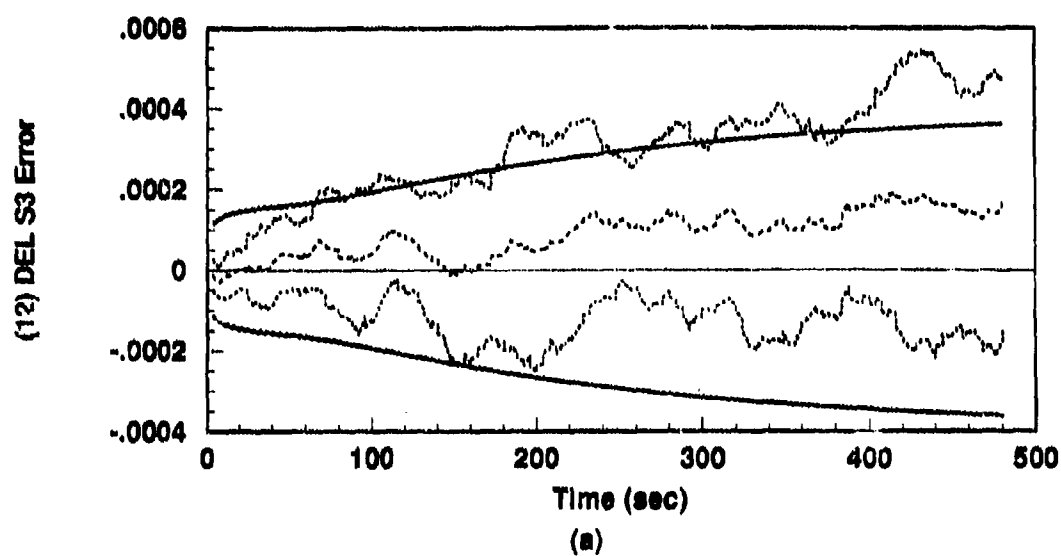
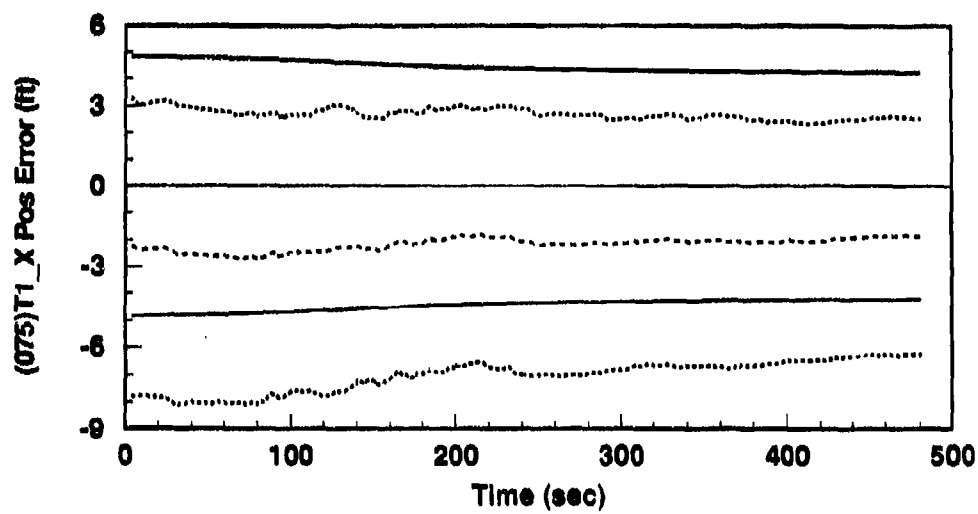
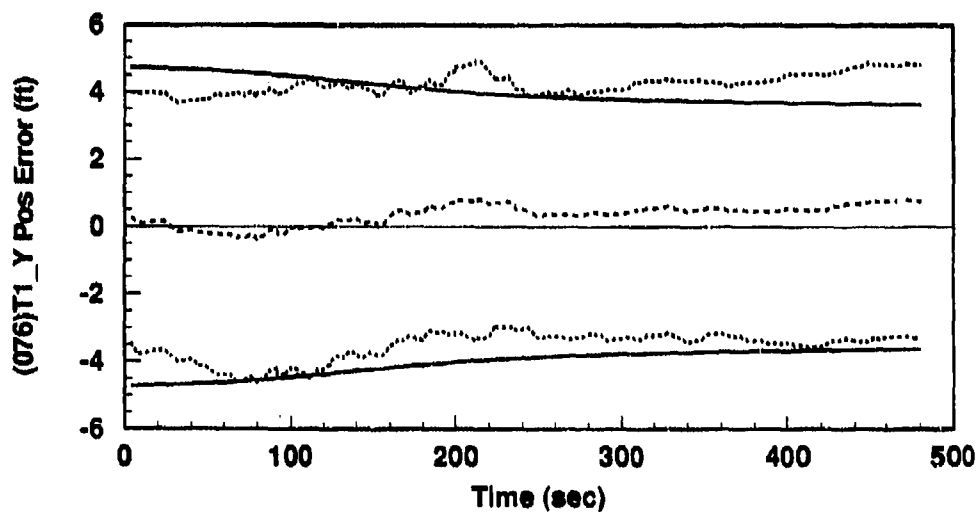


Figure G.6. Alignment: 98-State Model (a) ΔS_3 and (b) ΔS_4 Vertical Channel Aiding Error States. (Note: Initial conditions are those for Holloman AFB.)

----	Mean Error = $\bar{M}_x - (M_x)_{true}$
-----	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$



(a)



(b)

Figure G.7. Alignment: 98-State Model (a) Transponder 1, X Axis Position (b) Transponder 1, Y Axis Position Error States

----	$Mean\ Error = \hat{M}_x - (M_x)_{true}$
.....	$Mean\ Error \pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

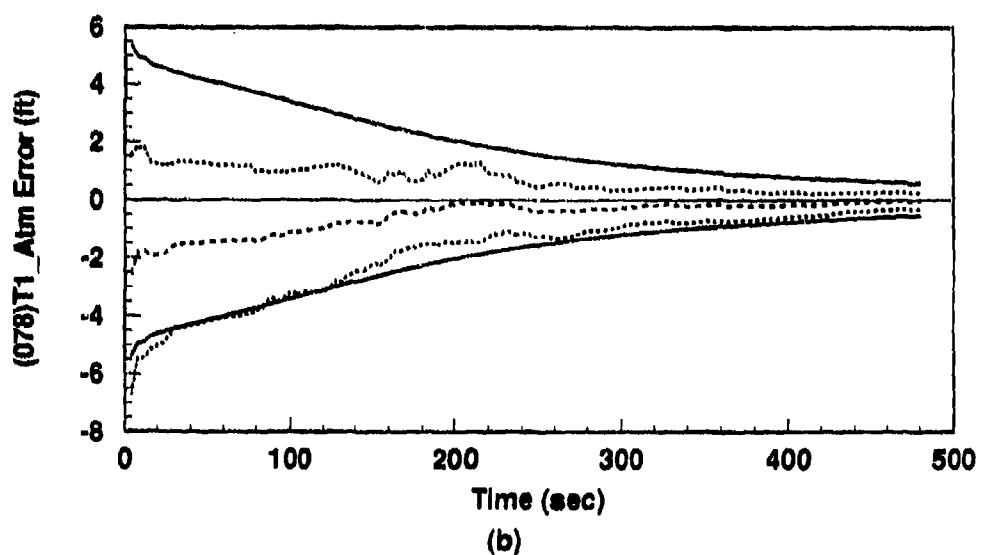
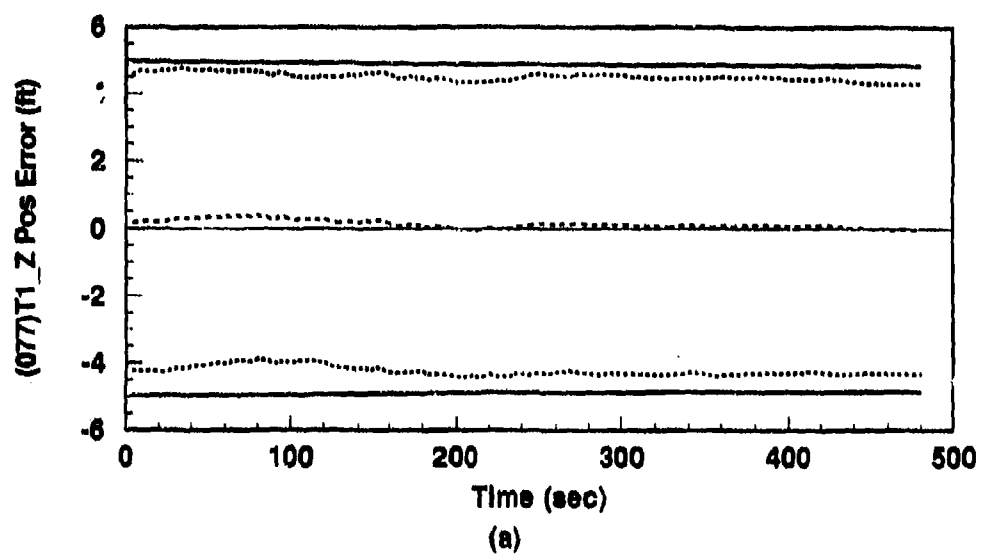


Figure G.8. Alignment: 98-State Model (a) Transponder 1, Z Axis Position (b) Transponder 1, Atmospheric Propagation Error States

----	$Mean\ Error = \hat{M}_x - (M_x)_{true}$
.....	$Mean\ Error \pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

G.2 98-State Model: Fighter Flight Using Holloman Initial Conditions

The plots in this section represent results of a 10-run flight simulation in which a flight profile (as described in Chapter III) is used to characterize the performance of the LN-93 with RRS aiding for a fighter mission originating and terminating at Holloman AFB, NM.

The purpose of this set of runs is to establish that the software function for the INS-RRS combination is comparable to (or better than) that achieved in the Litton flight runs reported in Appendices C through E. For this group of runs, baro-altimeter aiding is used in conjunction with RRS aiding.

The filter computed error estimates [σ_{filter}] are also compared to similar plots contained in the Litton reference documentation [20]. Comparisons are qualitative only and are intended to demonstrate trends. Such comparisons are contained in Chapter VI (Results).

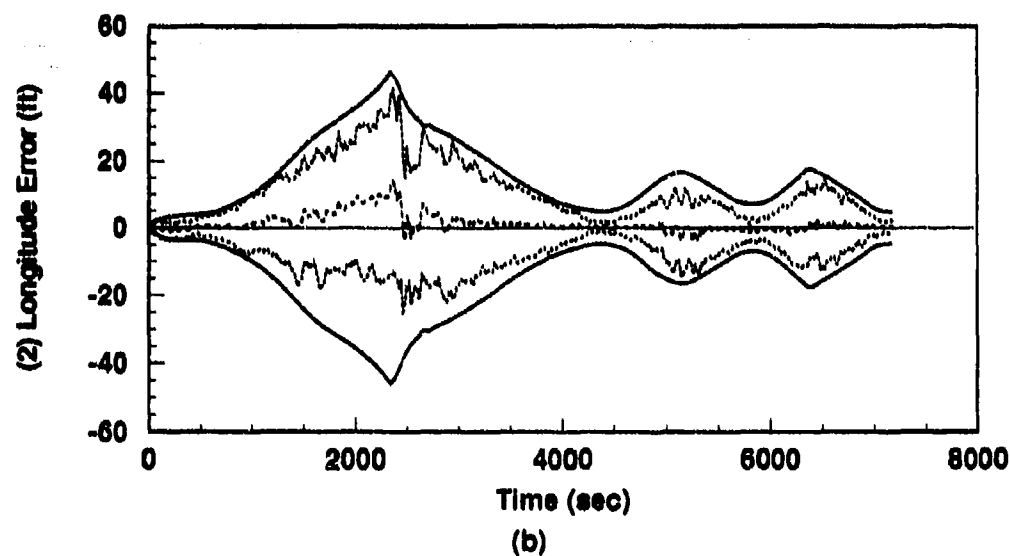
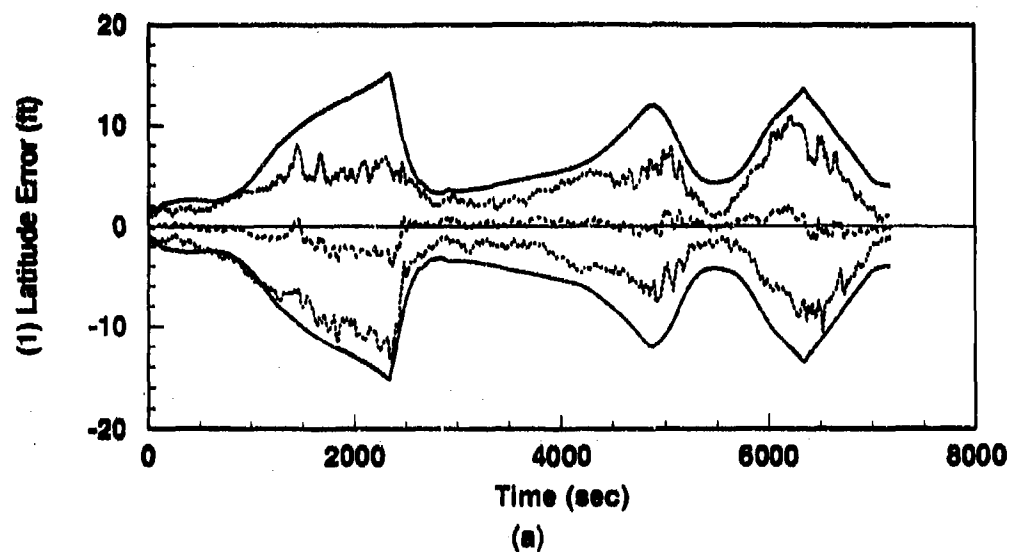


Figure G.9. Flight: 98-State Model (a) Latitude and (b) Longitude Error States. (Note: Initial conditions are those for Holloman AFB.)

---	Mean Error = $\bar{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

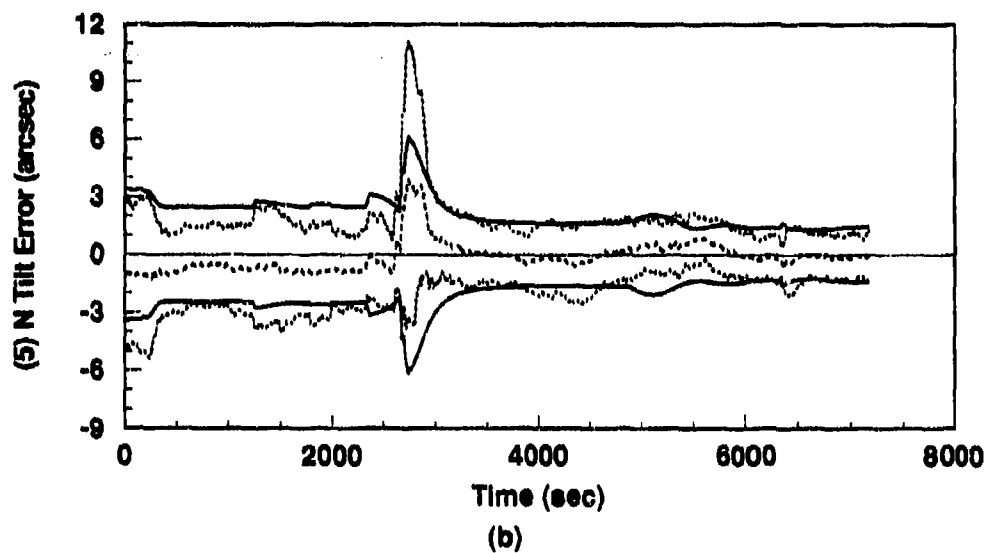
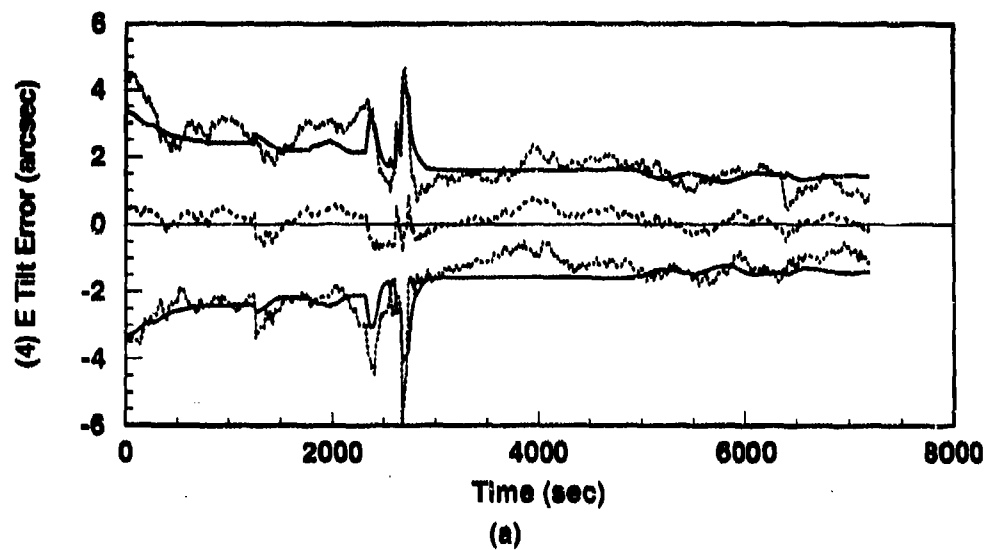


Figure G.10. Flight: 98-State Model (a) East Tilt and (b) North Tilt Error States. (Note: Initial conditions are those for Holloman AFB.)

----	Mean Error = $\bar{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

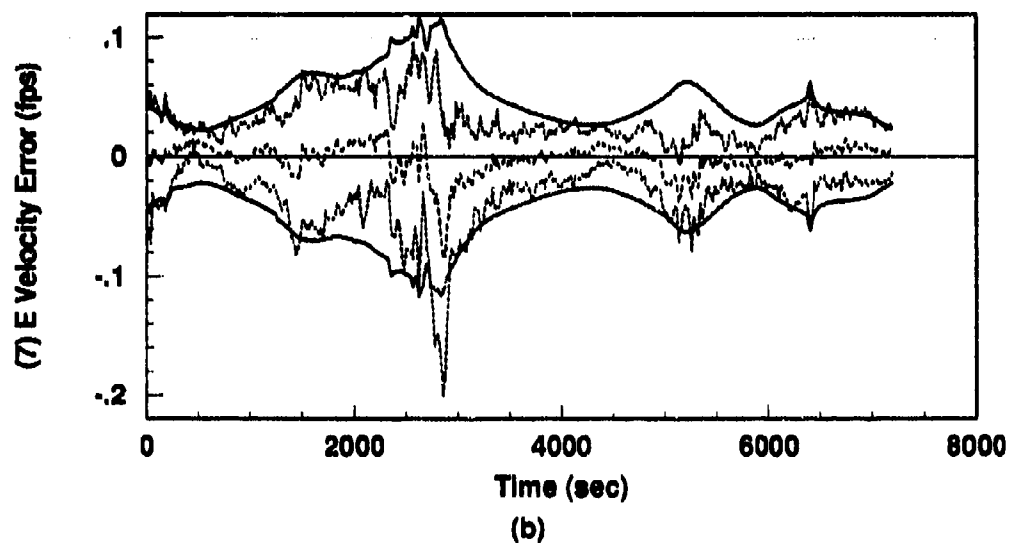
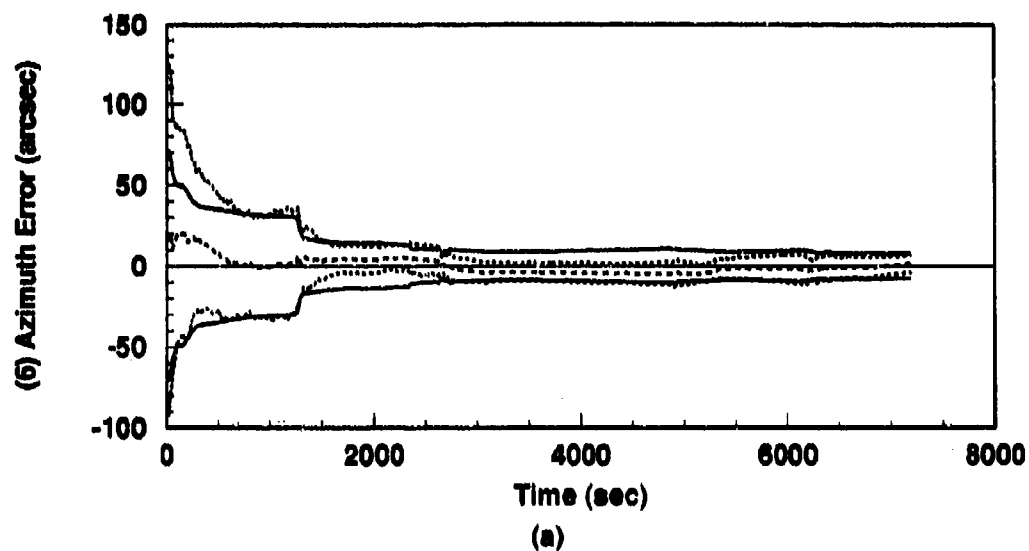


Figure G.11. Flight: 98-State Model (a) Azimuth and (b) East Velocity Error States.
(Note: Initial conditions are those for Holloman AFB.)

----	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

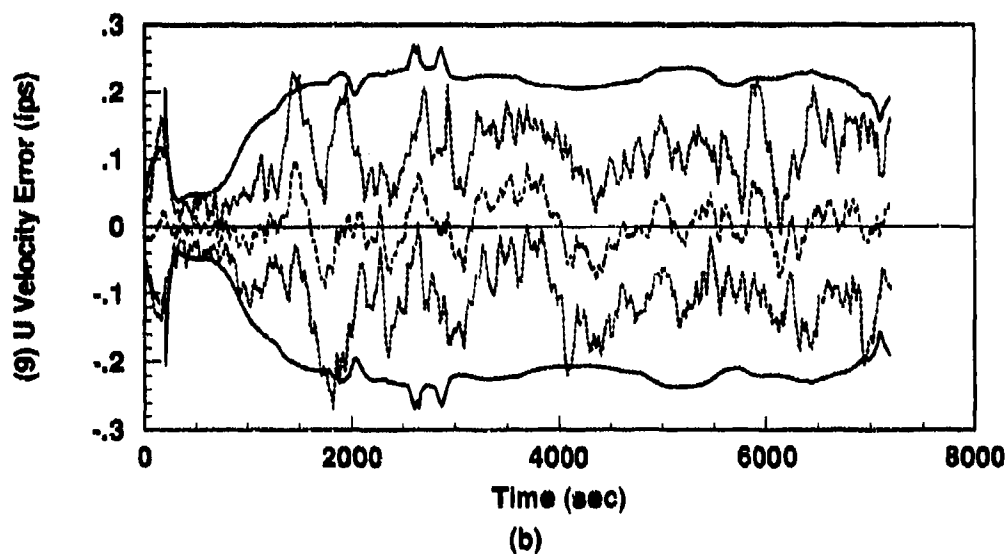
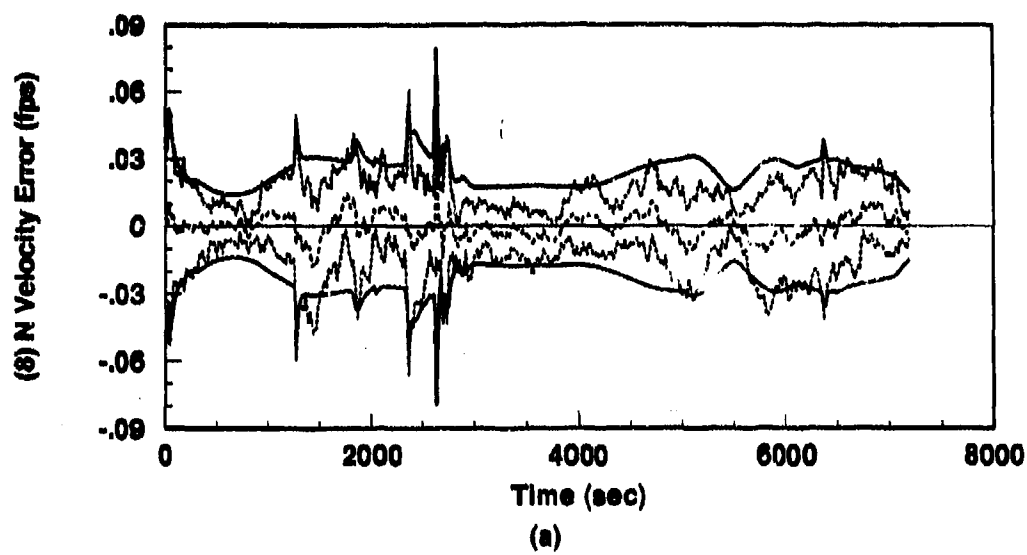


Figure G.12. Flight: 98-State Model (a) North Velocity and (b) Vertical Velocity Error States. (Note: Initial conditions are those for Holloman AFB.)

----	Mean Error = $\bar{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

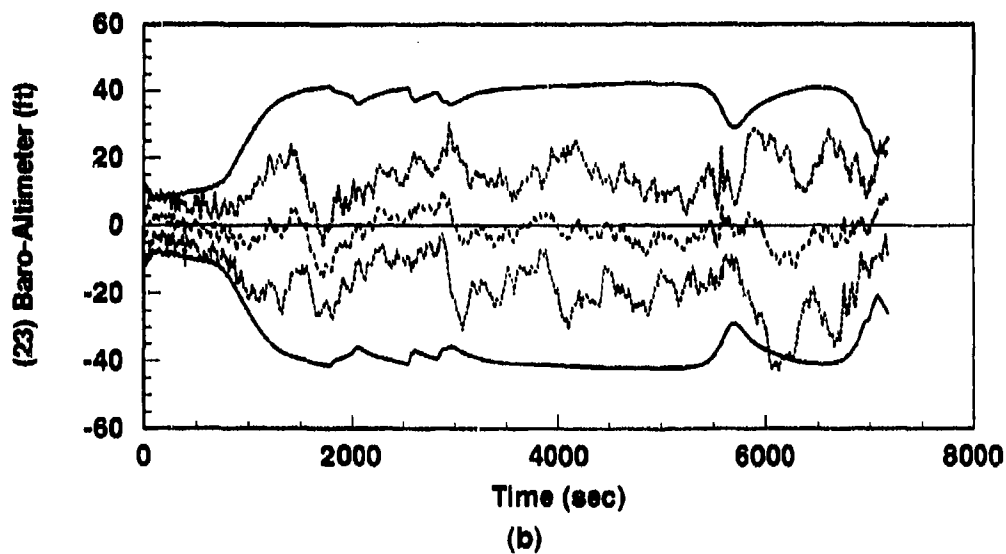
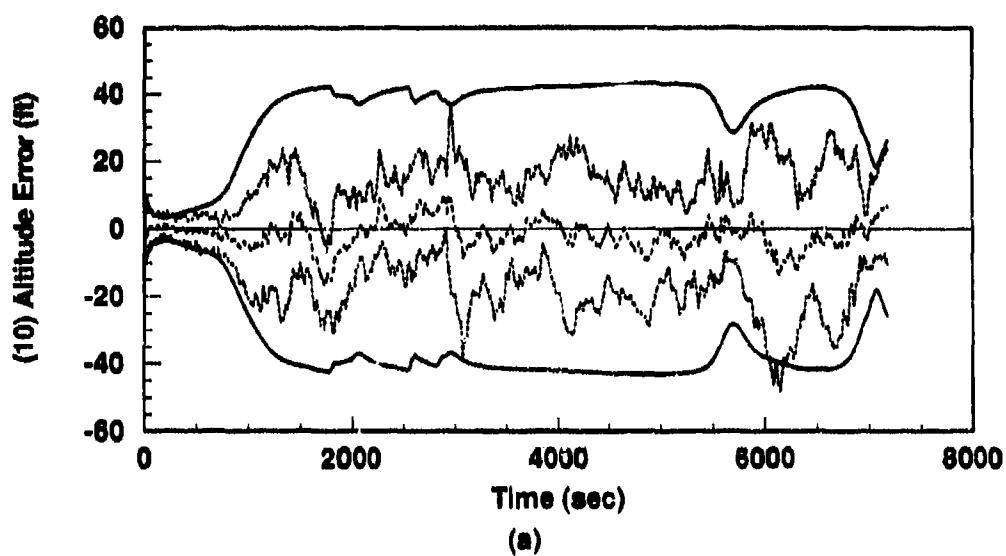


Figure G.13. Flight: 98-State Model (a) INS Altitude Error State and (b) Baro-Altimeter Total Error. (Note: Initial conditions are those for Holloman AFB.)

----	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

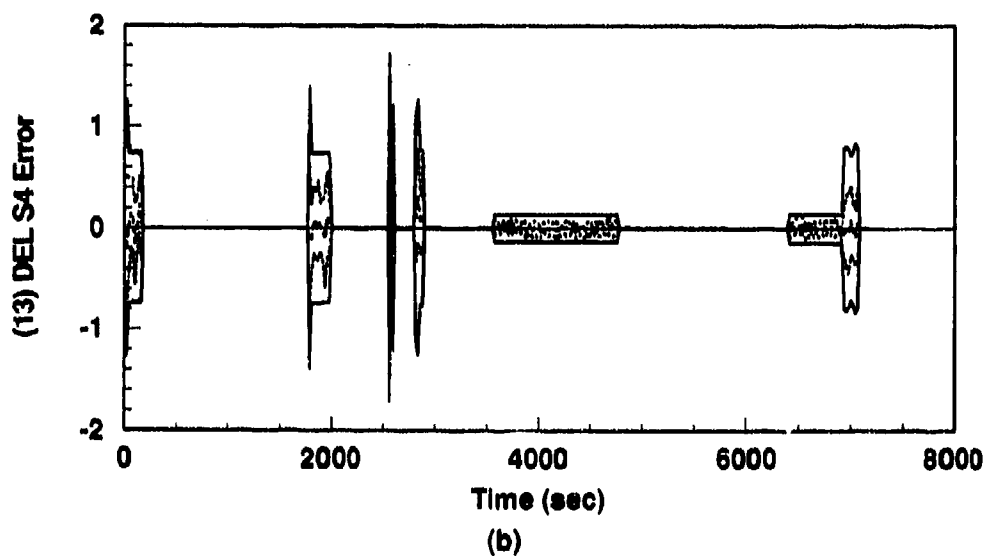
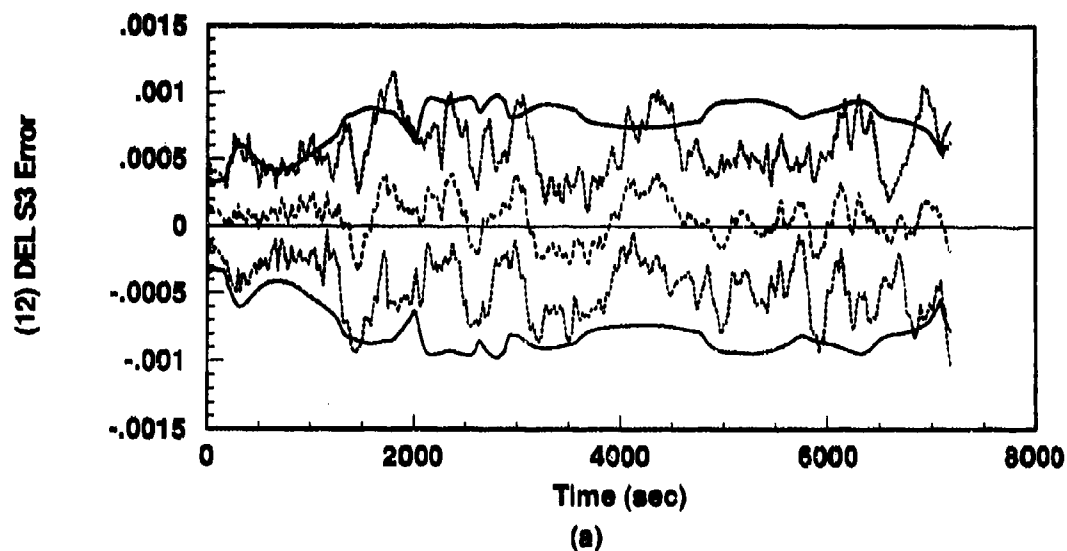


Figure G.14. Flight: 98-State Model (a) ΔS_3 and (b) ΔS_4 Vertical Channel Aiding Error States. (Note: Initial conditions are those for Holloman AFB.)

---	Mean Error = $\bar{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

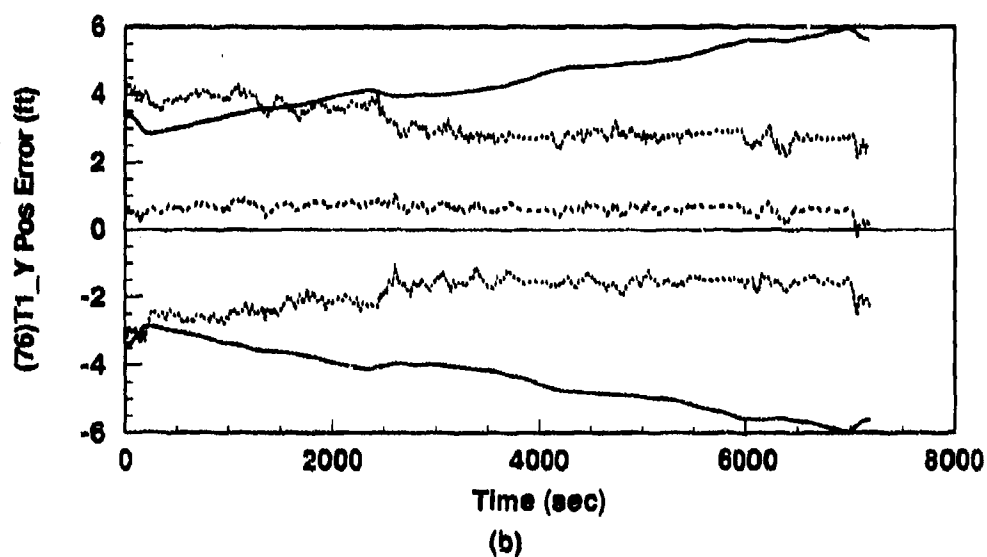
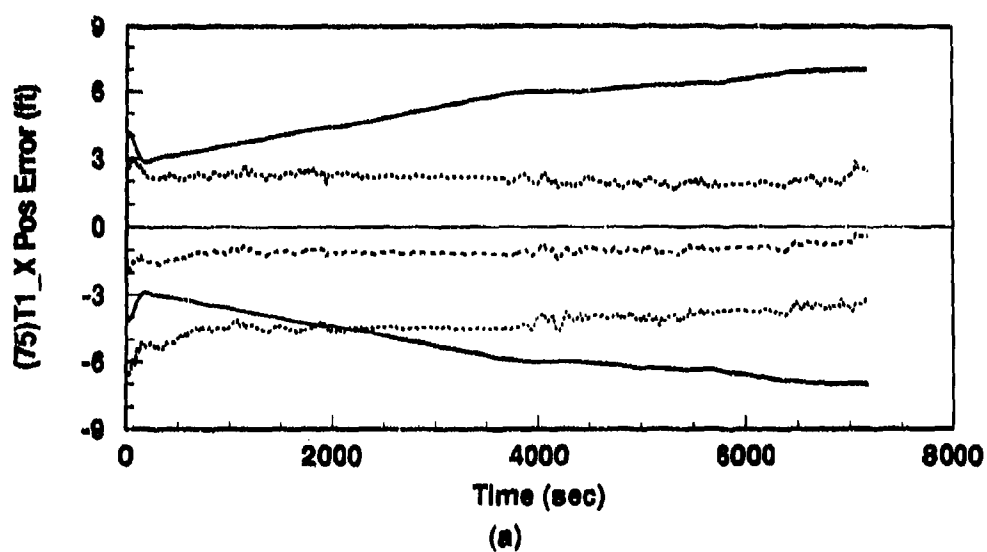


Figure G.15. Alignment: 98-State Model (a) Transponder 1, X Axis Position (b) Transponder 1, Y Axis Position Error States

----	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

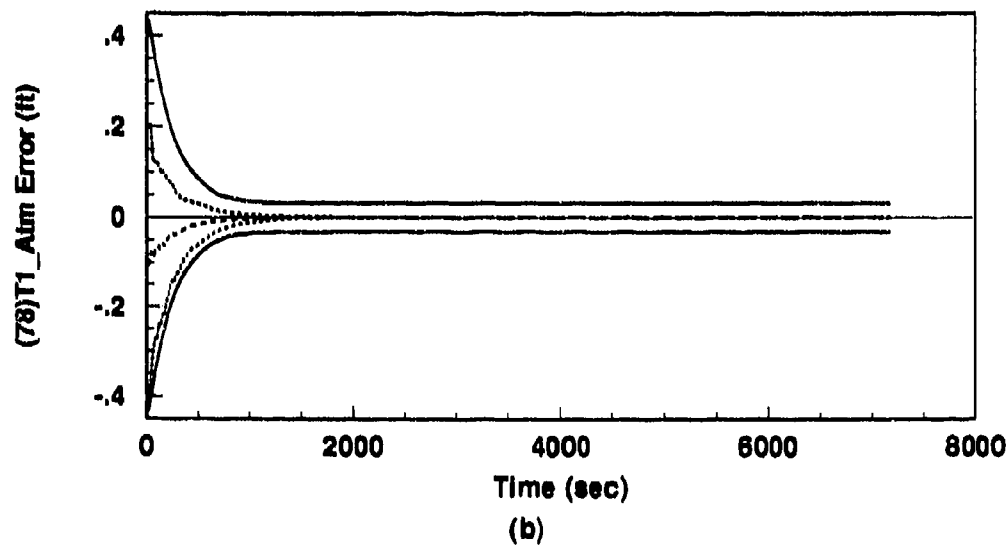
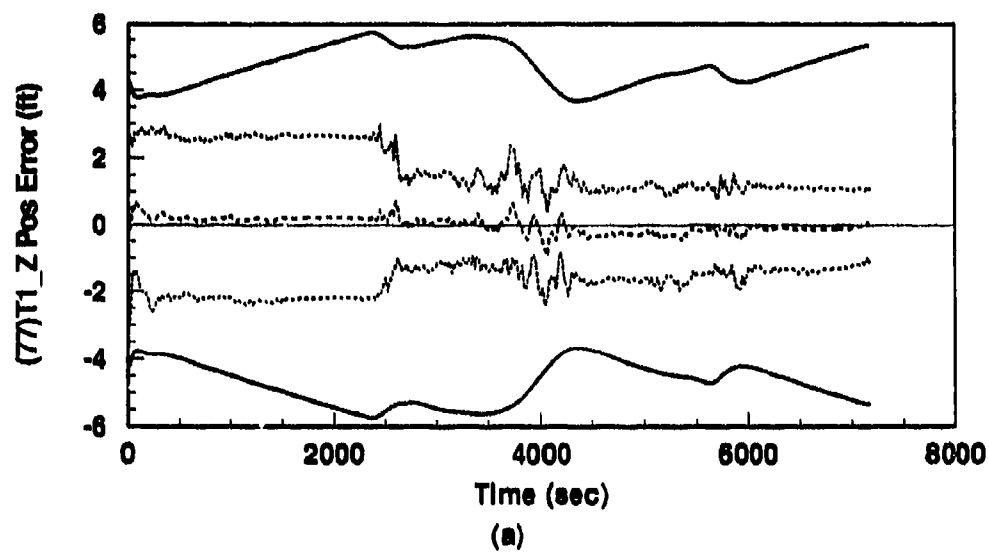


Figure G.16. Alignment: 98-State Model (a) Transponder 1, Z Axis Position (b) Transponder 1, Atmospheric Propagation Error States

----	Mean Error = $\bar{M}_T - (M_T)_{true}$
.....	Mean Error $\pm \sigma_{TRM}$
—	$0 \pm \sigma_{filter}$

Appendix H. 128-State INS-GPS Alignment and Flight Simulations

This appendix contains the plots from a 10-run Monte Carlo alignment and a 3-run Monte Carlo flight simulation of the 128-state error model which incorporates the truncated LN-93 INS, the revised baro-altimeter model, RRS, and GPS. Initial conditions are once again chosen for an alignment at Holloman AFB, NM. These results using GPS measurements only are compared to results from simulations in Appendices F, G, and I. RRS (transponder) measurements are NOT used in this configuration.

Variables plotted in this appendix are defined exactly in the same manner as those in Appendix C. All statistical quantities are calculated in the manner described at the beginning of Appendix C.

H.1 128-State Model: Alignment Using Holloman Initial Conditions

The plots in this section represent results of a 10-run Monte Carlo alignment simulation. In these alignments, INS aiding consists of velocity measurements, baro-altimeter measurements, and GPS pseudo-range measurements. (RRS measurements are NOT used.)

The purpose of this set of runs is to establish that the software function for the INS-GPS combination is comparable to (or better than) that achieved in the Litton flight runs reported in Appendices C through F.

Note that the plot for state 13, ΔS_4 is zero for all time during all alignment simulations. This is a normal condition due to the variable's dependence on altitude rate. The state becomes non-zero during flight runs. The GPS User clock states exhibit large magnitude transients during the initial phase of the alignment runs. As a result, an additional "window" is plotted on the page which follows the first user clock plots. The purpose is to demonstrate the steady state behavior of these critical error states. All other plots are discussed in Chapter VI (Results).

The filter computed error estimates $[\sigma_{filter}]$ are also compared to similar plots contained in the Litton reference documentation [20]. Comparisons are qualitative only and

are intended to demonstrate trends. Such comparisons are contained in Chapter VI (Results).

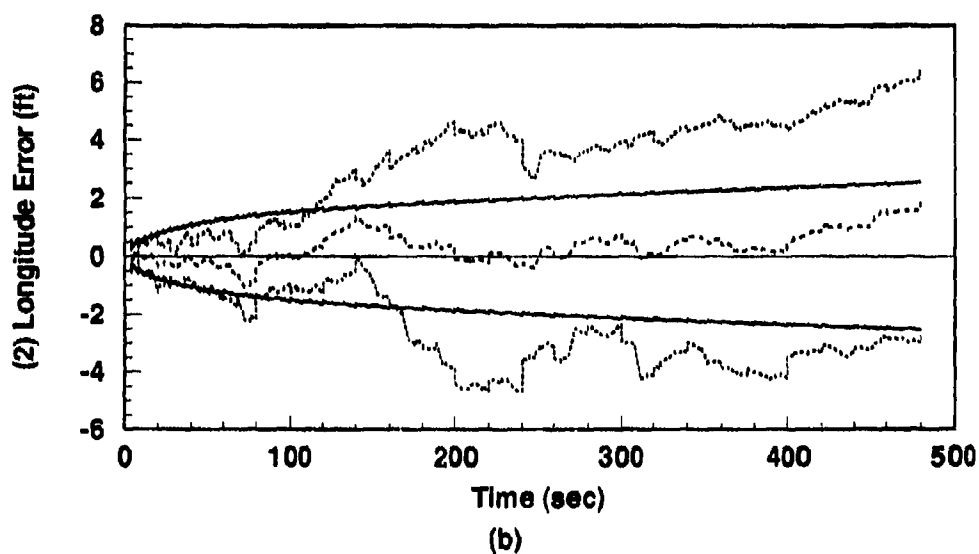
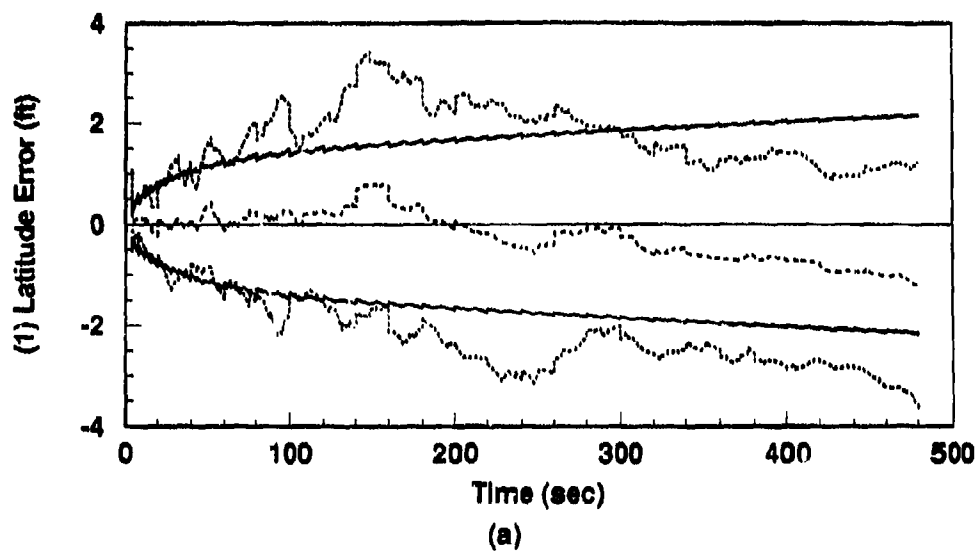


Figure H.1. Alignment: 128-State Model (a) Latitude and (b) Longitude Errors. (Note: GPS Measurements Only)

---	$Mean\ Error = \hat{M}_x - (M_x)_{true}$
.....	$Mean\ Error \pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

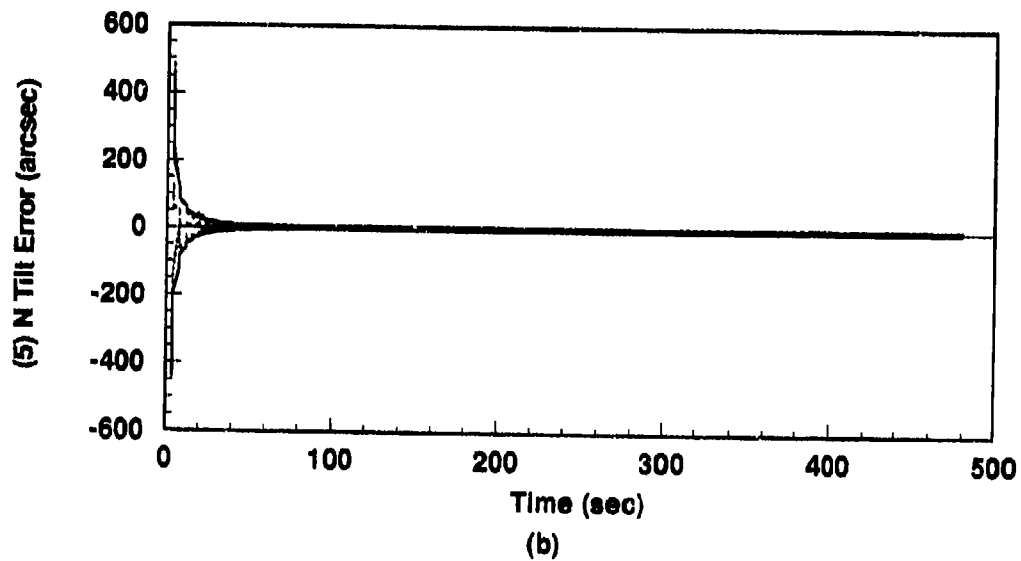
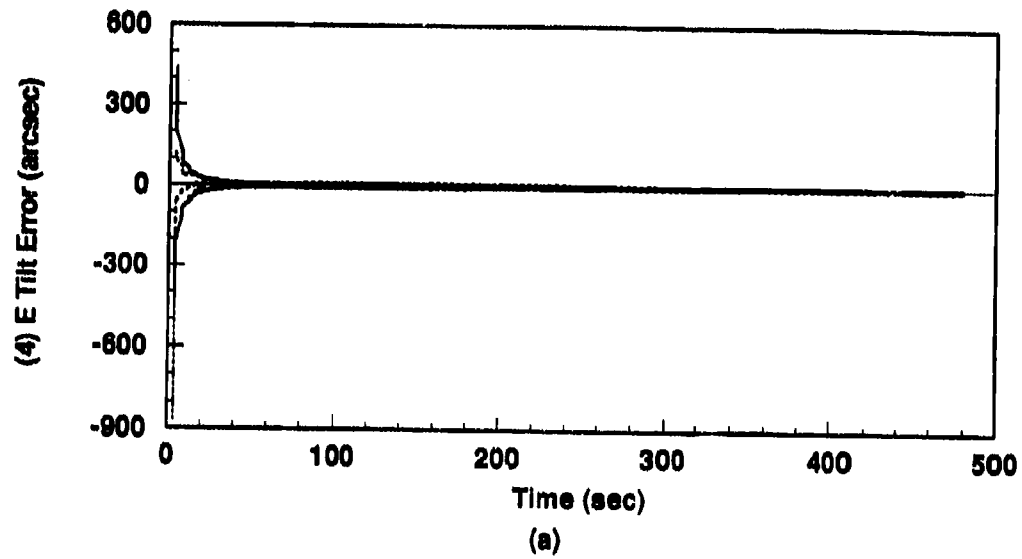


Figure H.2. Alignment: 128-State Model (a) East Tilt and (b) North Tilt Error States.
(Note: GPS Measurements Only)

----	Mean Error = $\bar{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
-----	$0 \pm \sigma_{filter}$

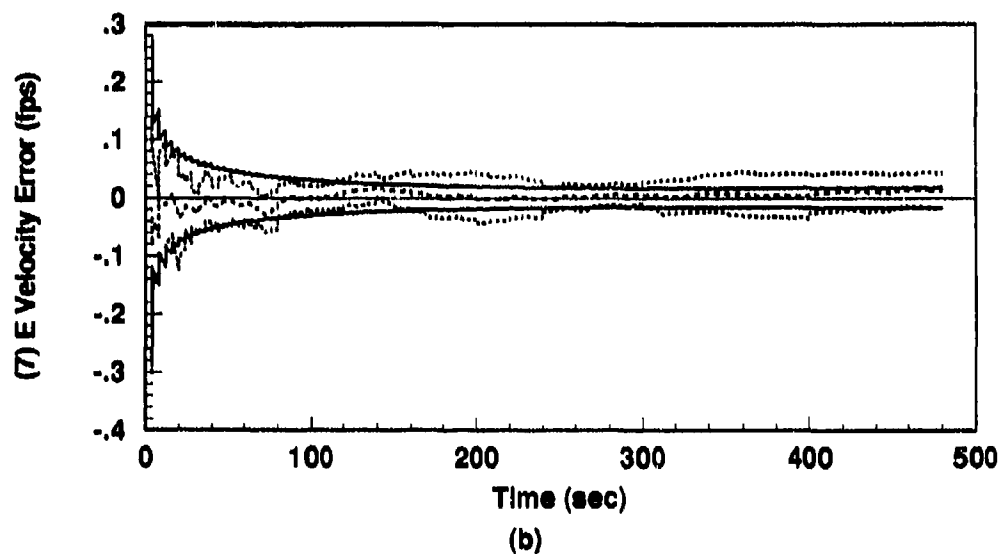
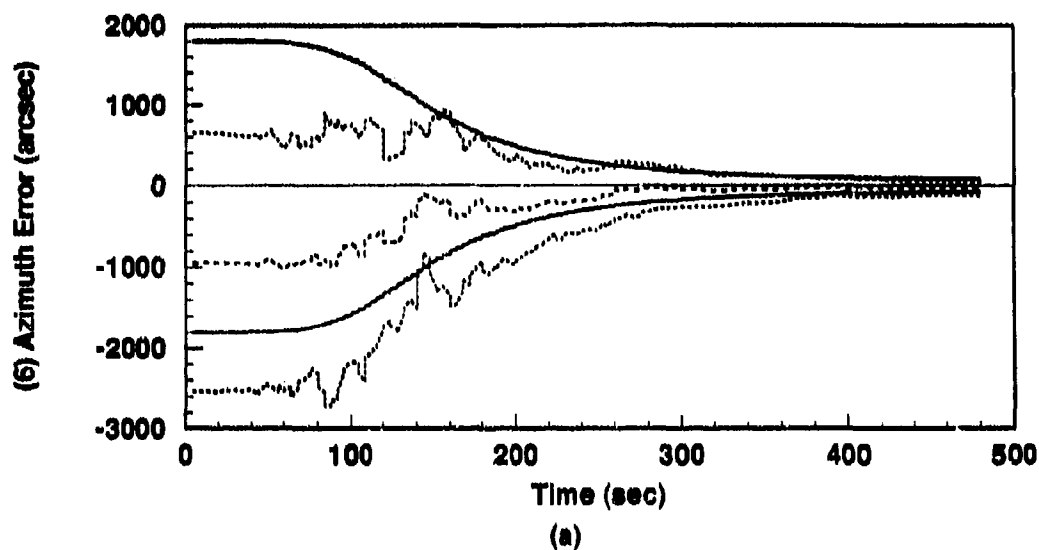


Figure H.3. Alignment: 128-State Model (a) Azimuth and (b) East Velocity Error States.
(Note: GPS Measurements Only)

----	Mean Error = $\bar{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
----	$0 \pm \sigma_{filter}$

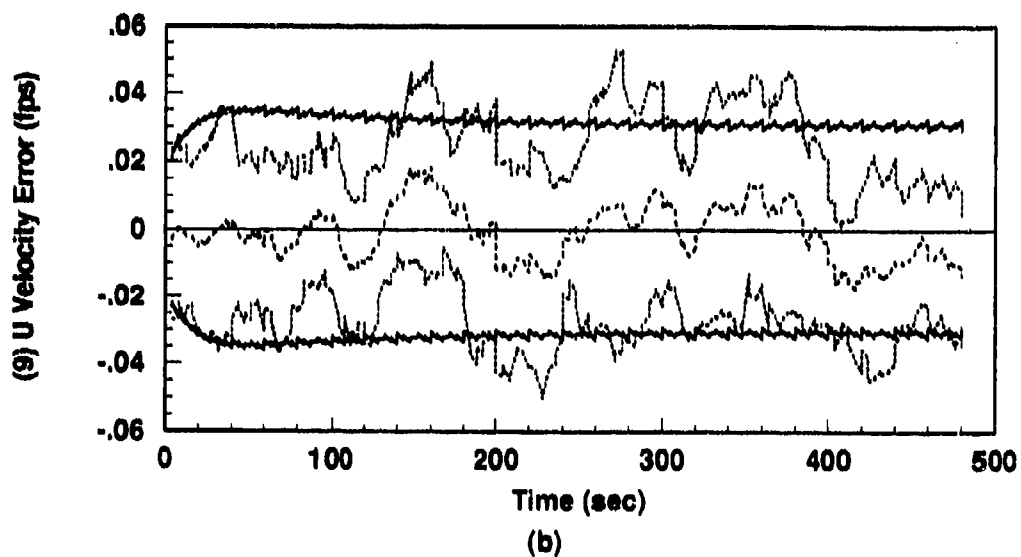
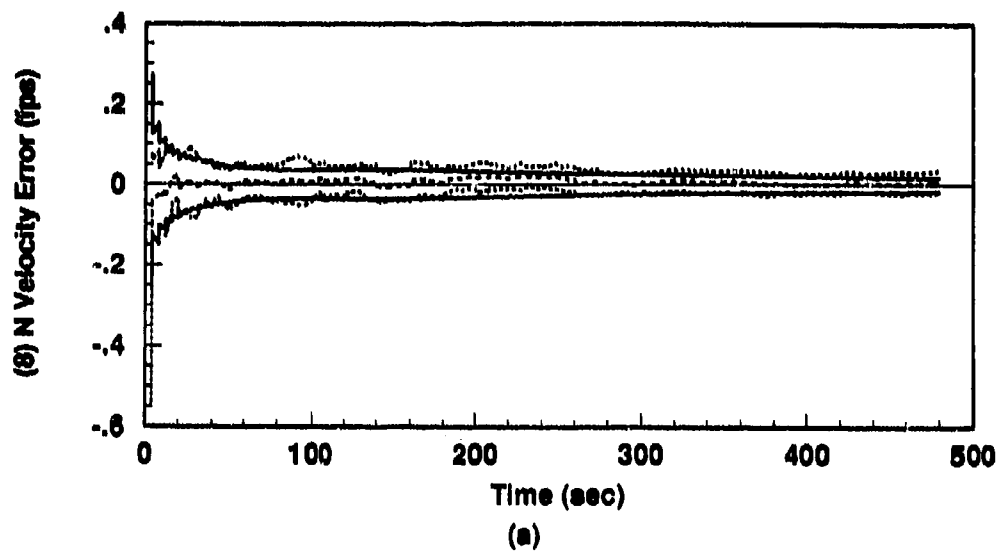


Figure H.4. Alignment: 128-State Model (a) North Velocity and (b) Vertical Velocity Error States. (Note: GPS Measurements Only)

---	$Mean\ Error = \hat{M}_x - (M_x)_{true}$
.....	$Mean\ Error \pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

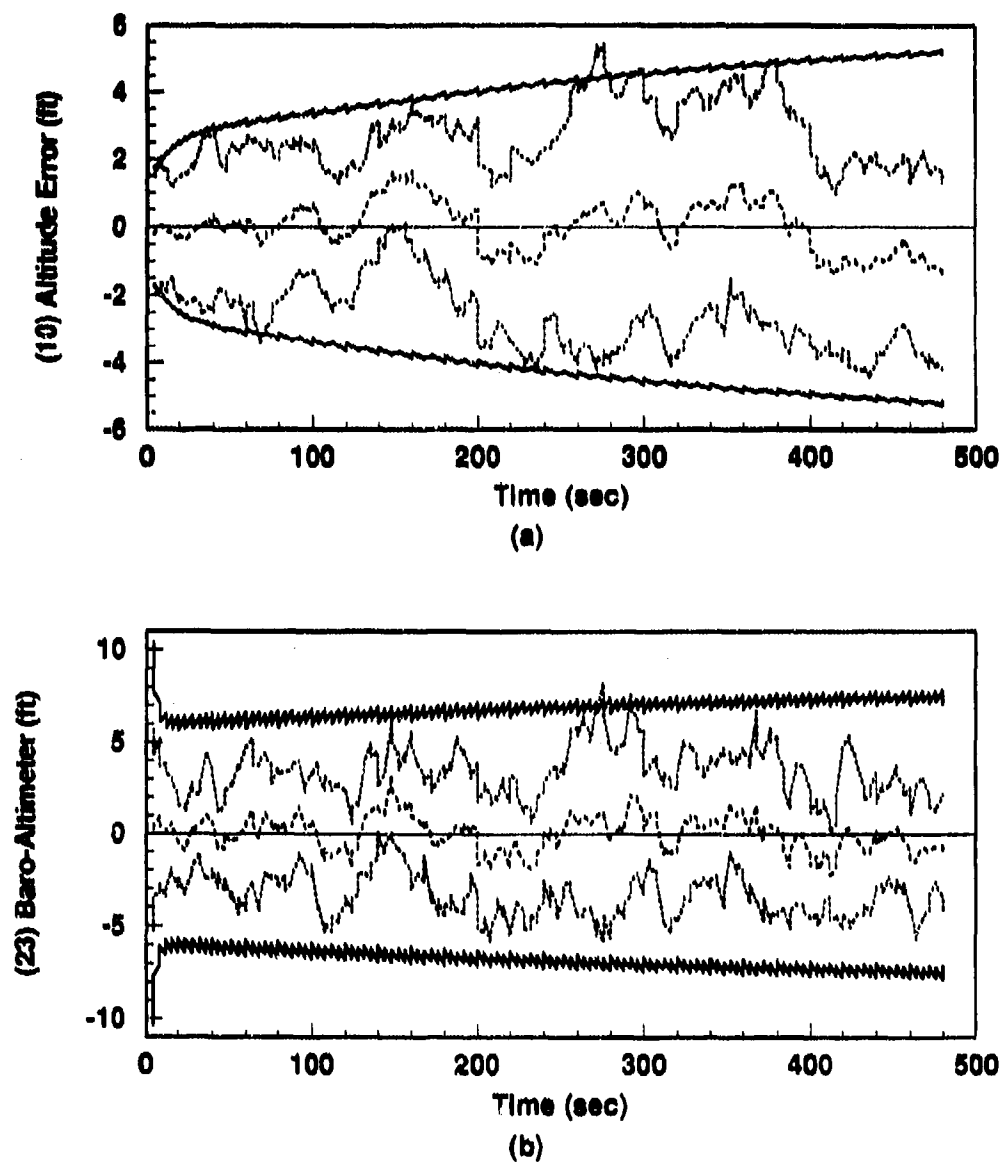


Figure H.5. Alignment: 128-State Model (a) INS Altitude Error State and (b) Baro-Altimeter Total Error. (Note: GPS Measurements Only)

---	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
---	$0 \pm \sigma_{filter}$

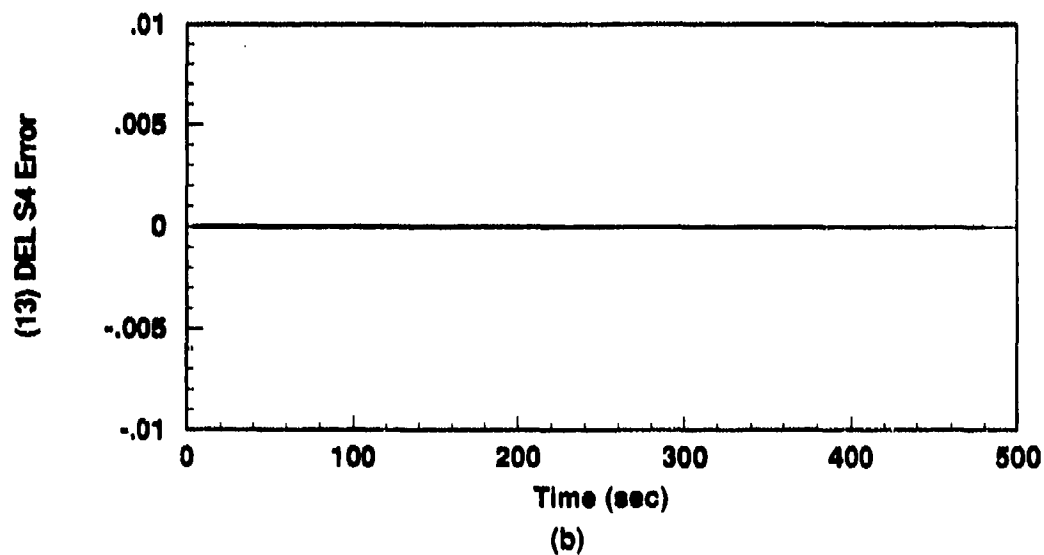
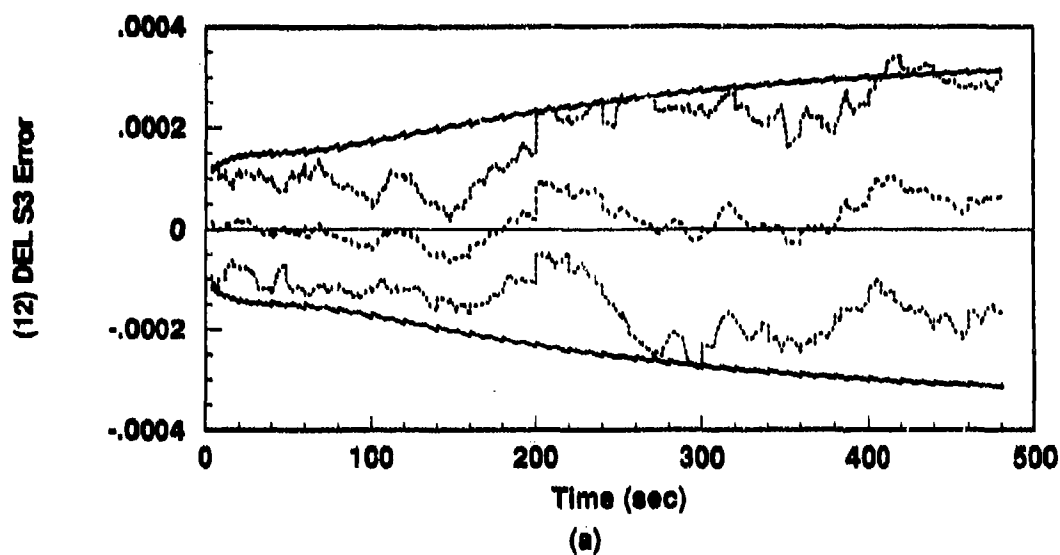


Figure H.6. Alignment: 128-State Model (a) ΔS_1 and (b) ΔS_1 Vertical Channel Aiding Error States. (Note: GPS Measurements Only)

----	$Mean\ Error = \hat{M}_x - (M_x)_{true}$
.....	$Mean\ Error \pm \sigma_{trim}$
—	$0 \pm \sigma_{filter}$

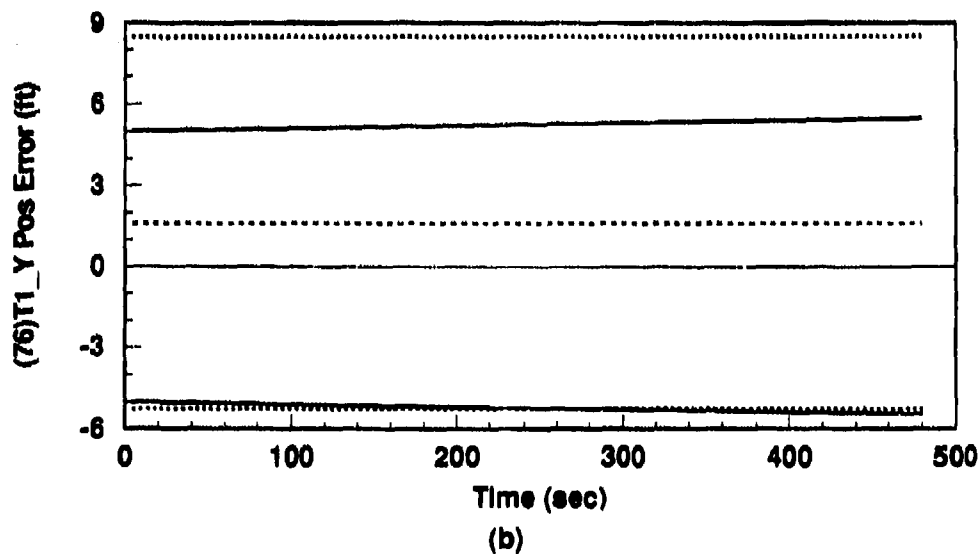
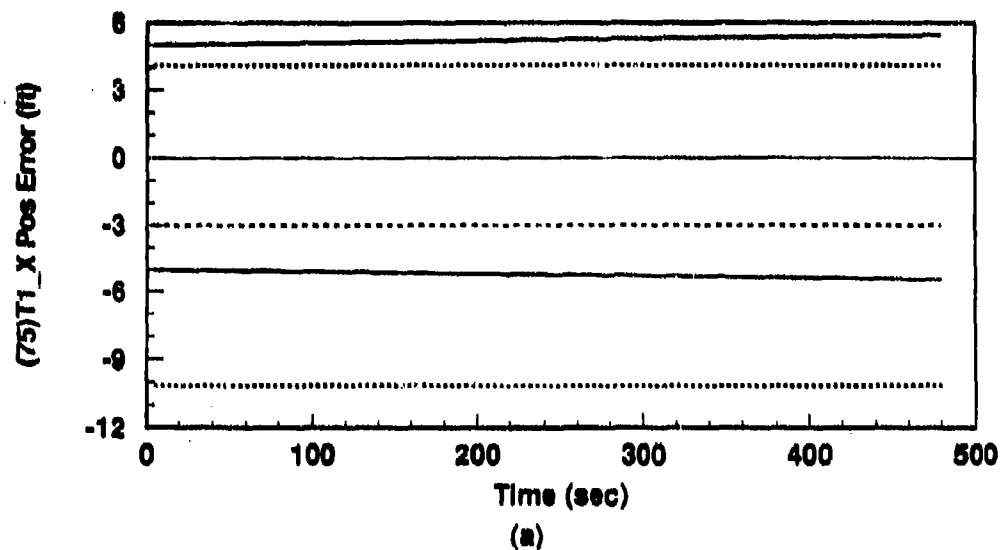


Figure H.7. Alignment: 128-State Model (a) Transponder 1, X Axis Position (b) Transponder 1, Y Axis Position Error States (Note: GPS Measurements Only)

.....	$Mean\ Error = \bar{M}_x - (M_x)_{true}$
.....	$Mean\ Error \pm \sigma_{true}$
———	$0 \pm \sigma_{filter}$

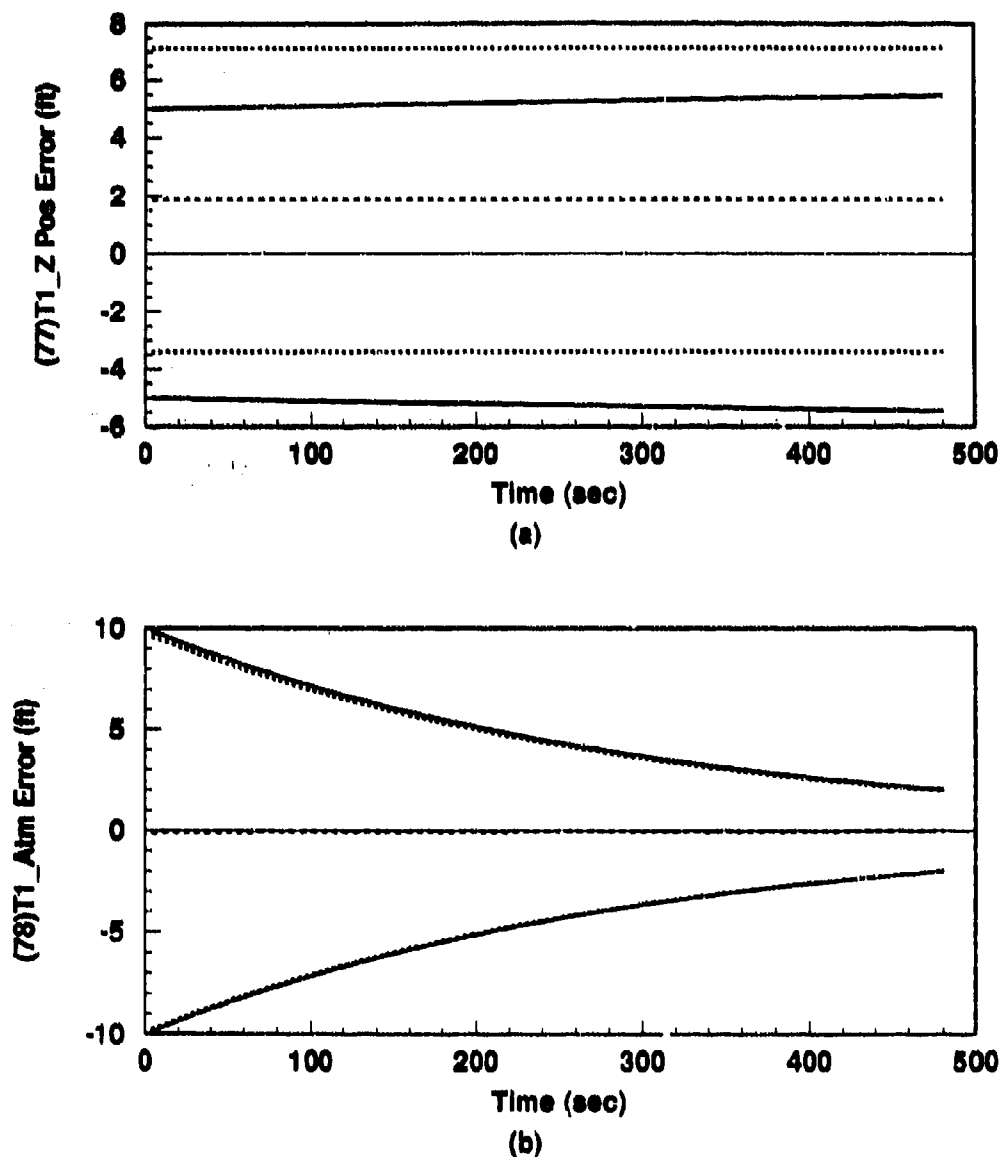


Figure H.8. Alignment: 128-State Model (a) Transponder 1, Z Axis Position (b) Transponder 1, Atmospheric Propagation Error States (Note: GPS Measurements Only)

----	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
-----	$0 \pm \sigma_{filter}$

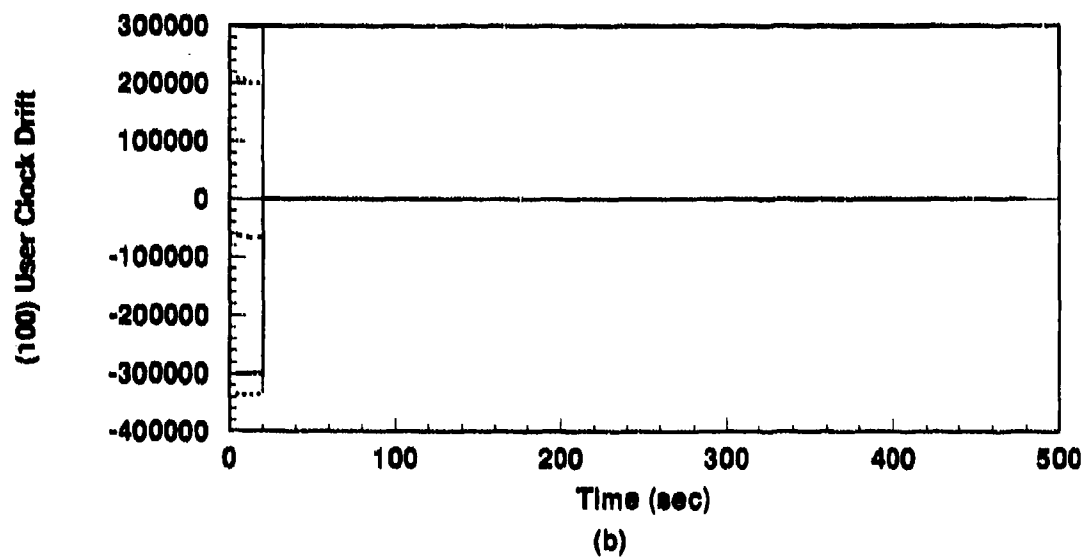
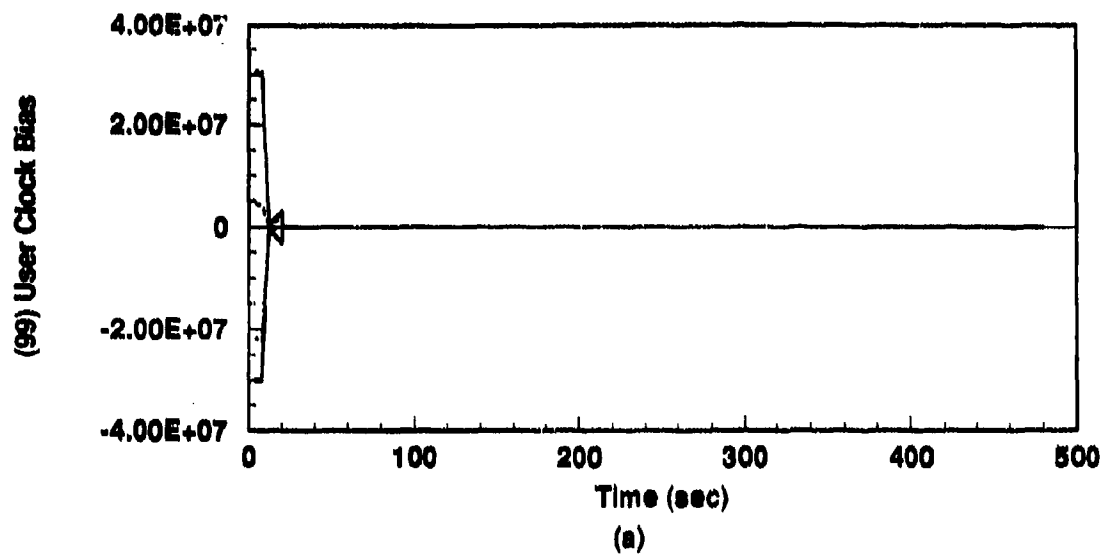


Figure H.9. Alignment: 128-State Model (a) User Clk Bias and (b) User Clock Drift States (Note: GPS Measurements Only)

....	$Mean\ Error = \hat{M}_x - (M_x)_{true}$
.....	$Mean\ Error \pm \sigma_{true}$
----	$0 \pm \sigma_{filter}$

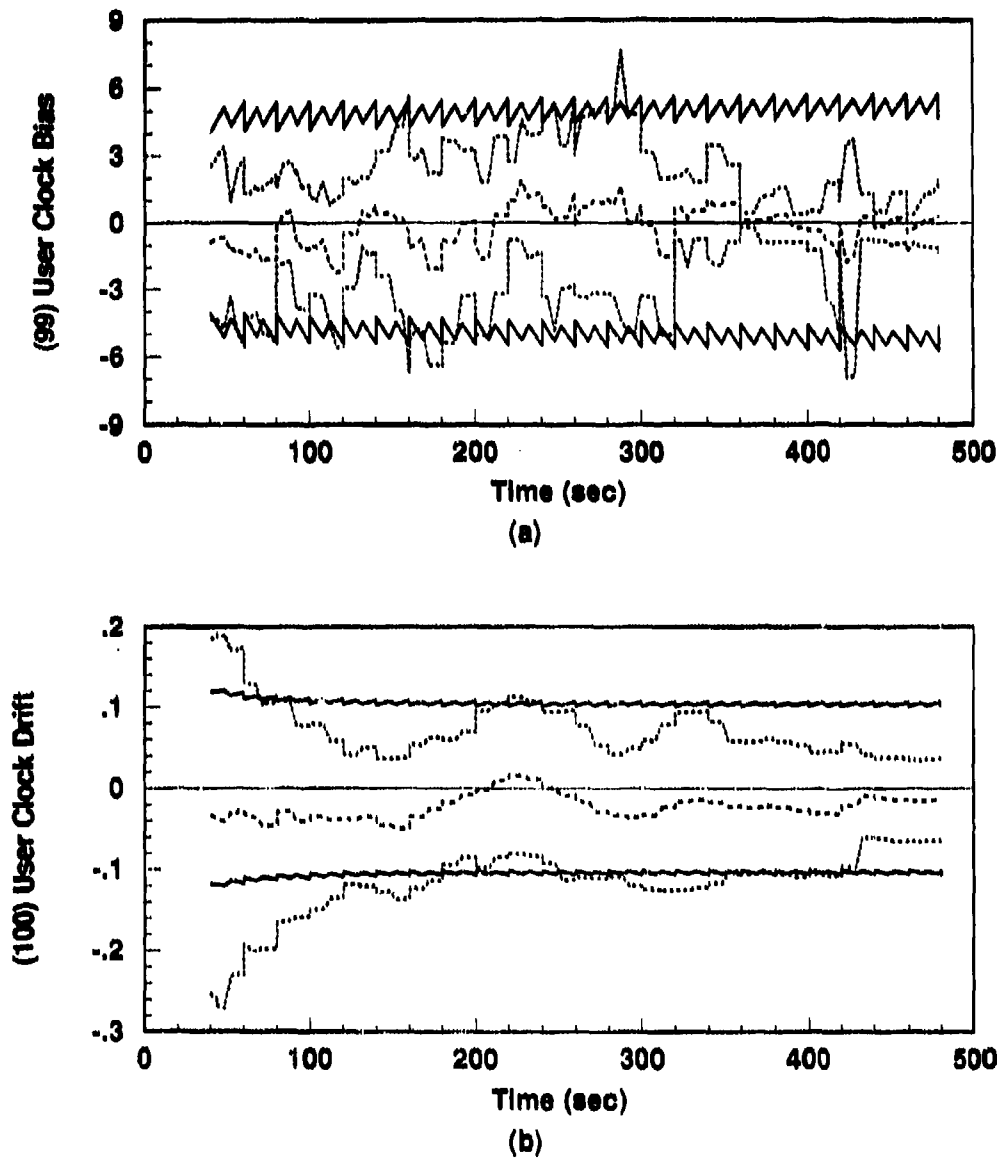


Figure H.10. Alignment: 128-State Model: Steady State (a) User Clk Bias and (b) User Clock Drift States (Note: GPS Measurements Only)

----	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

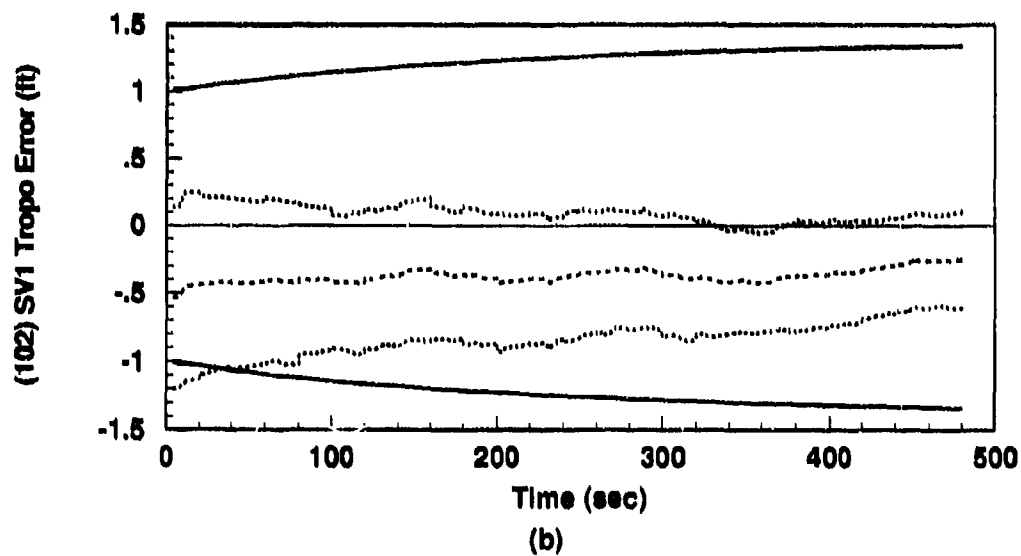
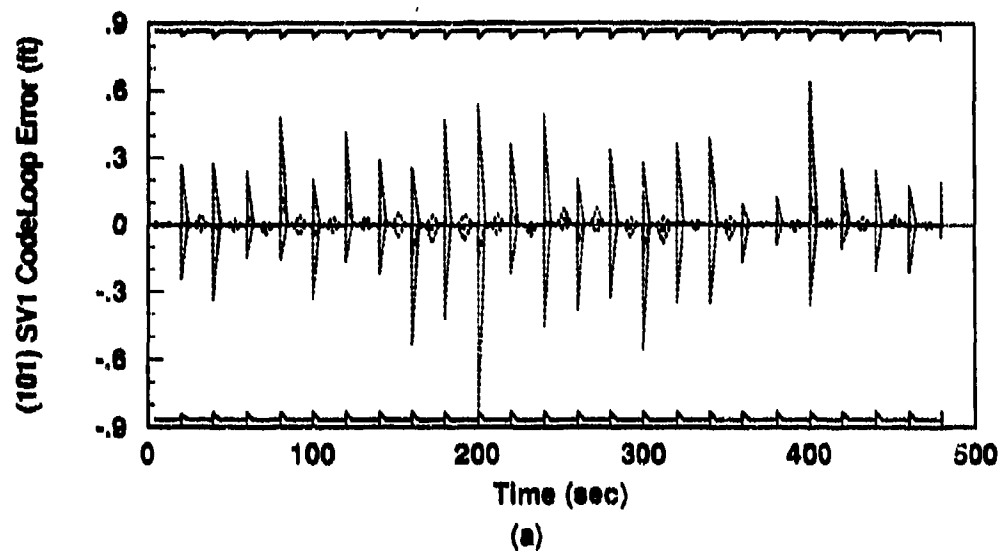


Figure H.11. Alignment: 128-State Model (a) SV 1, CodeLoop and (b) SV 1, Atmospheric Propagation Error States (Note: GPS Measurements Only)

- - - -	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
————	$0 \pm \sigma_{filter}$

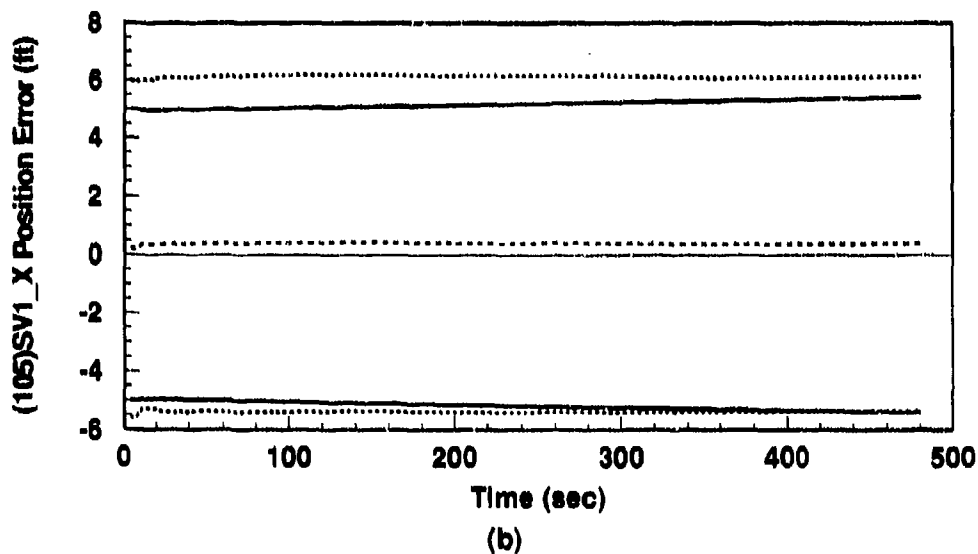
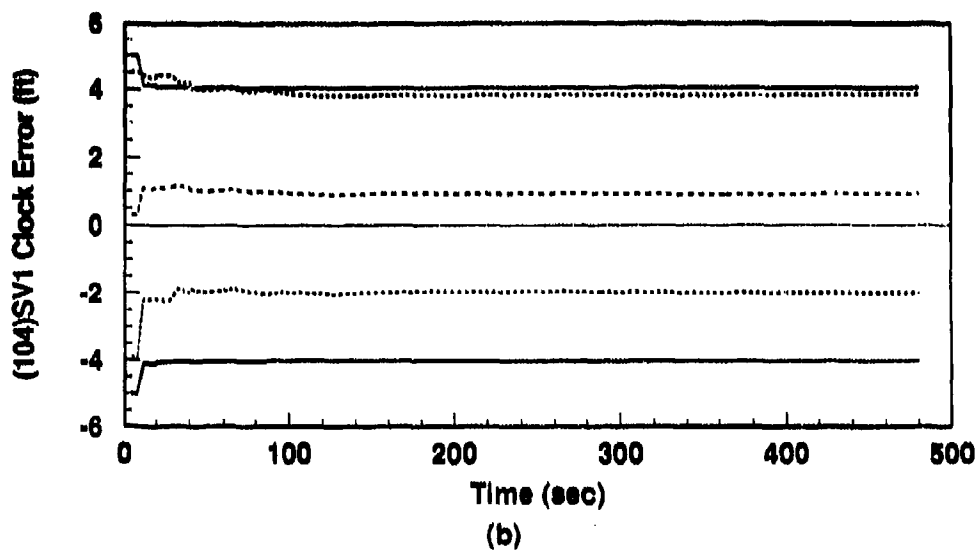


Figure H.12. Alignment: 128-State Model (a) SV 1, Clock and (b) SV 1, X Position Error States (Note: GPS Measurements Only)

.....	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
-----	$0 \pm \sigma_{filter}$

H.2 128-State Model: Fighter Flight Using Holloman Initial Conditions

The plots in this section represent results of a 3-run flight simulation in which a flight profile (as described in Chapter III) is used to characterize the performance of the LN-93 with RRS aiding for a fighter mission originating and terminating at Holloman AFB, NM.

It is well understood that a 3-run simulation has reduced statistical validity [22, 23, 24]. However, the execution time for a 10-run simulation of this size is prohibitive. Therefore, the 3-run results are included for a CAUTIOUS comparison.

The purpose of this set of runs is to establish that the software function for the INS-RRS-GPS combination is comparable to (or better than) that achieved in the Litton flight runs reported in Appendices C through F.

The filter computed error estimates [σ_{filter}] are also compared to similar plots contained in the Litton reference documentation [20]. Comparisons are qualitative only and are intended to demonstrate trends. Such comparisons are contained in Chapter VI (Results).

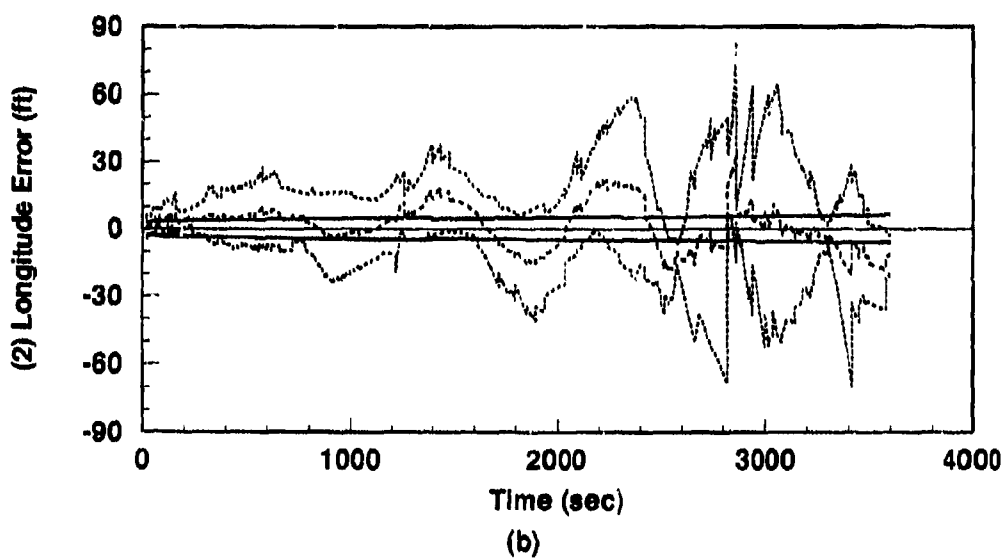
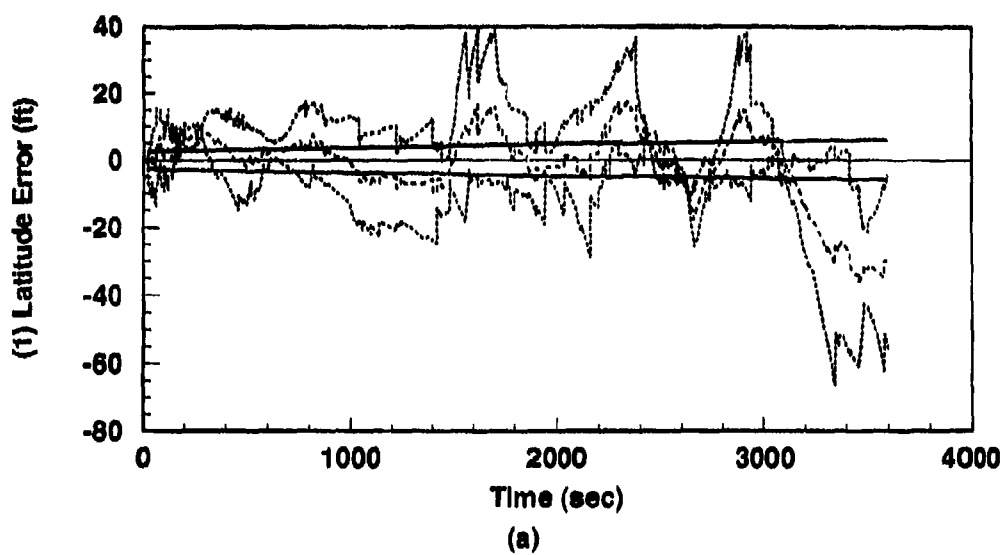
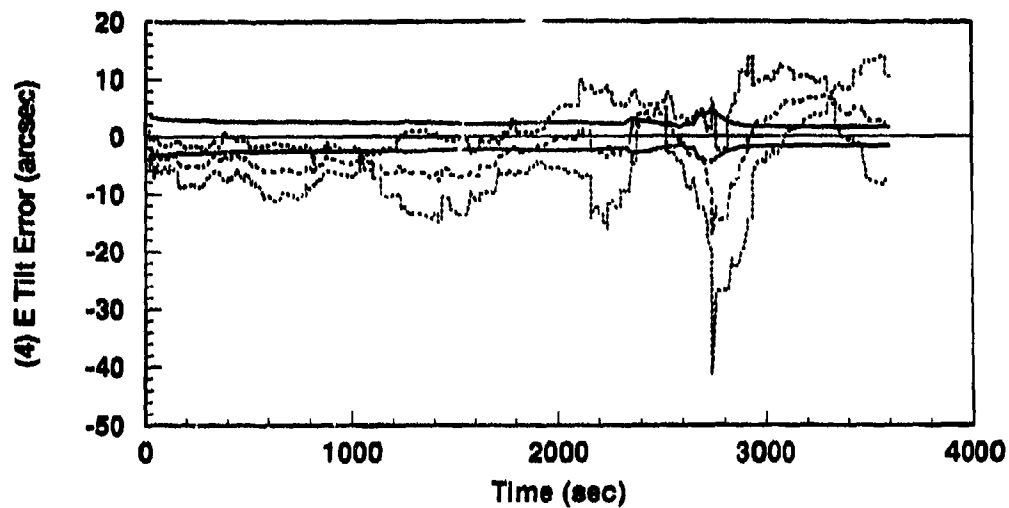
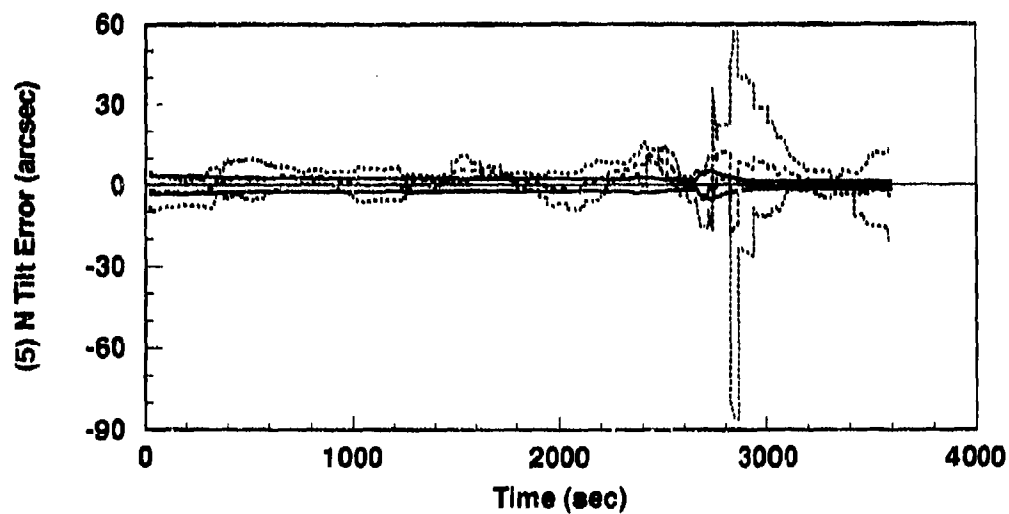


Figure H.13. Flight: 128-State Model (a) Latitude and (b) Longitude Error States.
(Note: GPS Measurements Only)

----	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$



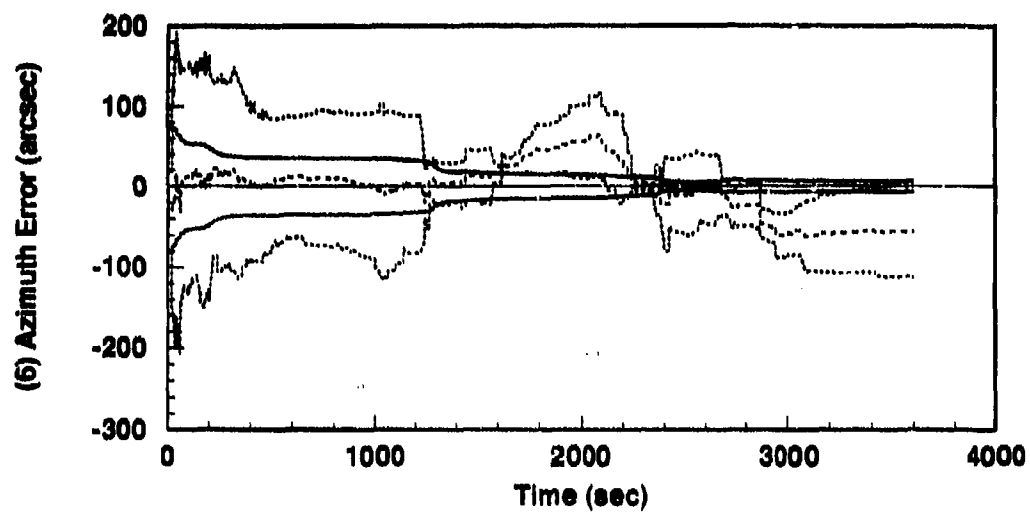
(a)



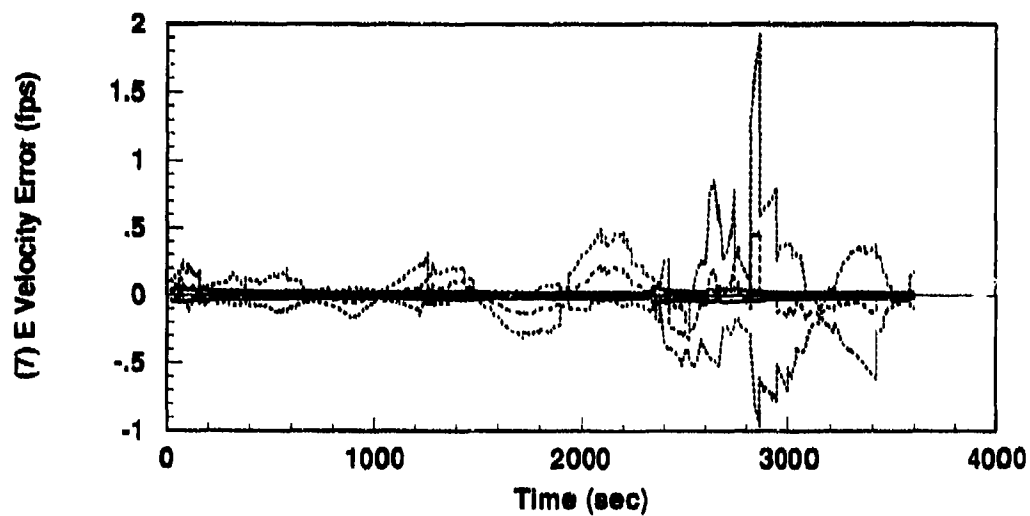
(b)

Figure H.14. Flight: 128-State Model (a) East Tilt and (b) North Tilt Error States.
(Note: GPS Measurements Only)

----	Mean Error = $\bar{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$



(a)



(b)

Figure H.15. Flight: 128-State Model (a) Azimuth and (b) East Velocity Error States.
(Note: GPS Measurements Only)

----	$Mean\ Error = \hat{M}_x - (M_x)_{true}$
.....	$Mean\ Error \pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

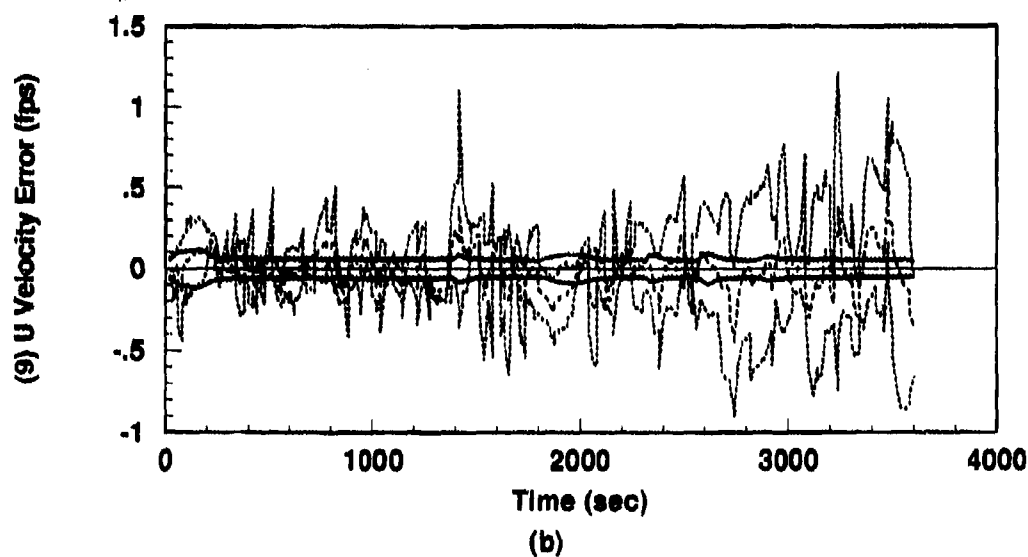
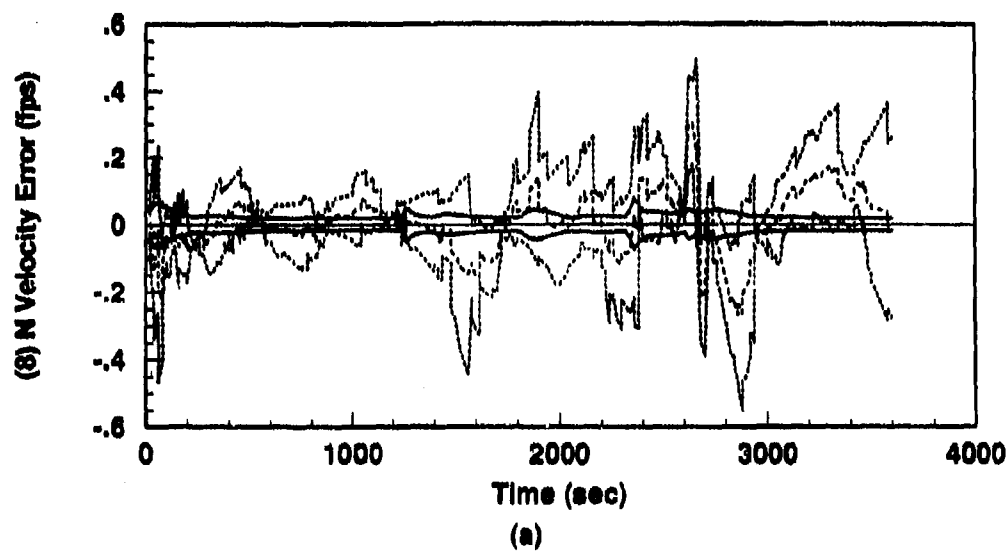


Figure H.16. Flight: 128-State Model (a) North Velocity and (b) Vertical Velocity Error States. (Note: GPS Measurements Only)

----	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

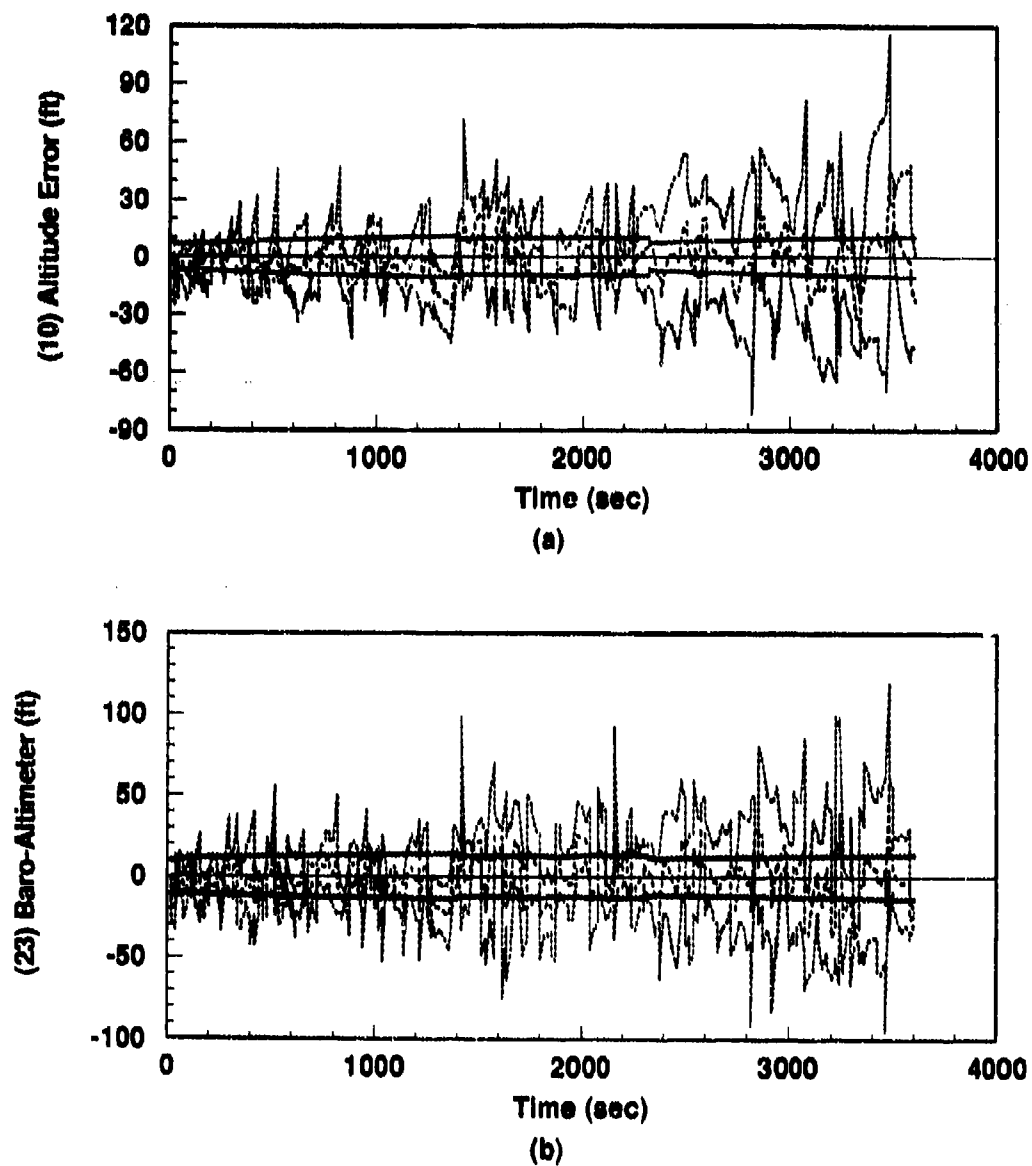


Figure H.17. Flight: 128-State Model (a) INS Altitude Error State and (b) Baro-Altimeter Total Error. (Note: GPS Measurements Only)

---	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{trim}$
—	$0 \pm \sigma_{filter}$

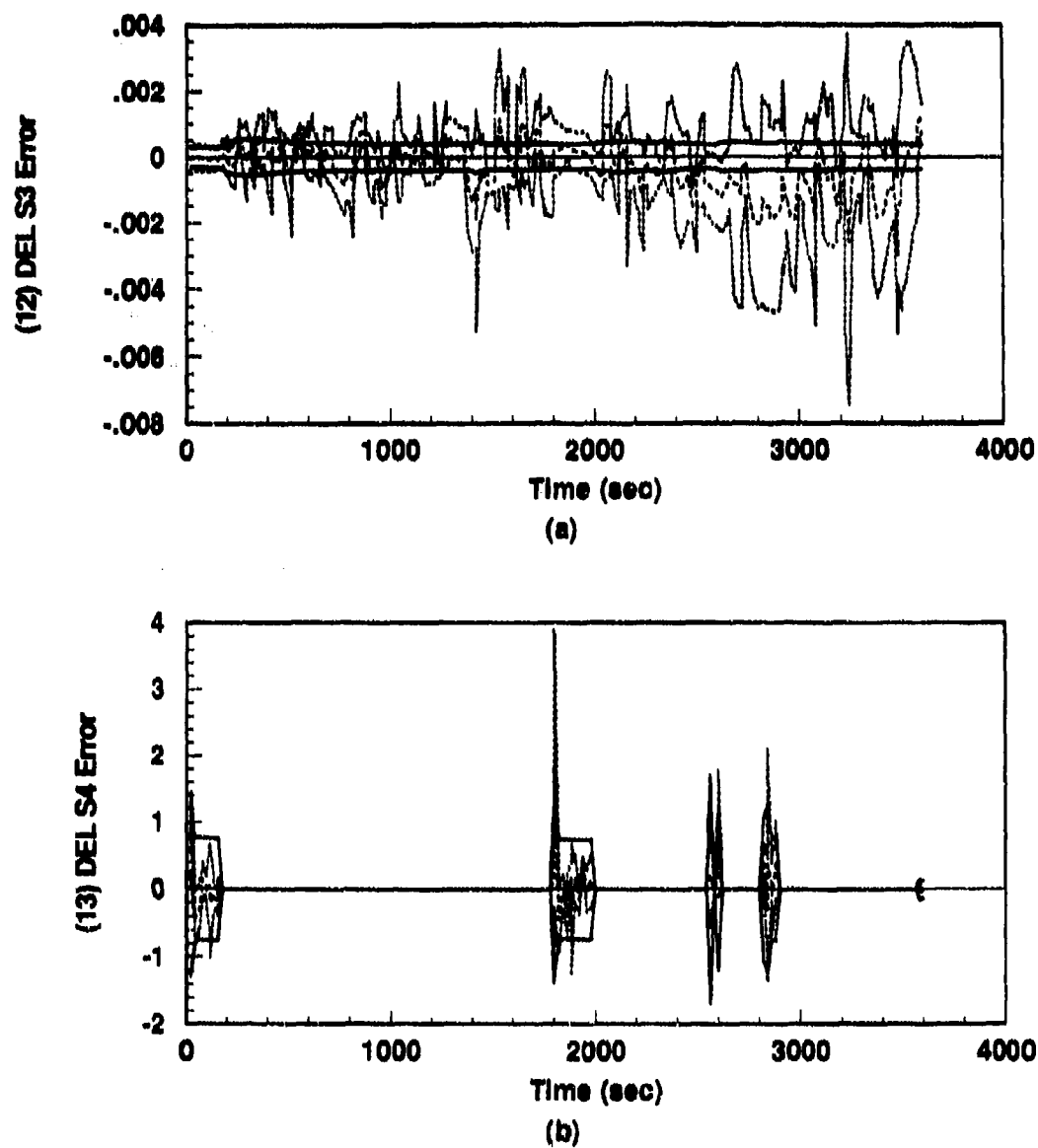


Figure H.18. Flight: 128-State Model (a) ΔS_3 and (b) ΔS_4 Vertical Channel Aiding Error States. (Note: GPS Measurements Only)

....	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

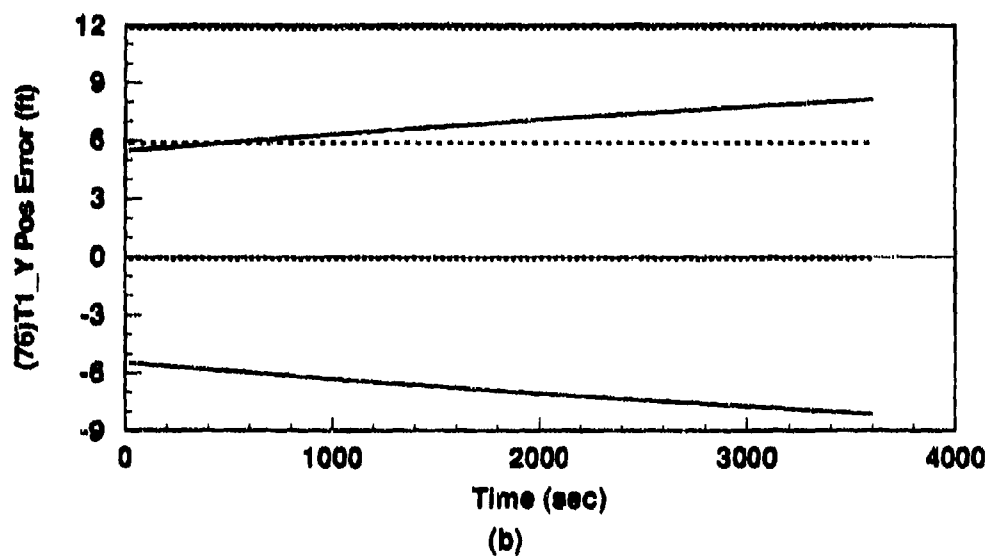
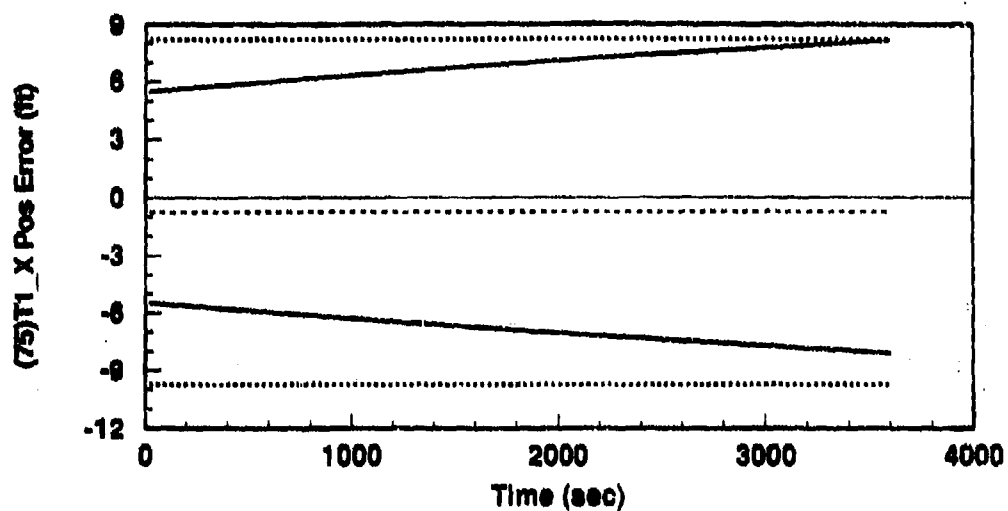


Figure H.19. Alignment: 128-State Model (a) Transponder 1, X Axis Position (b) Transponder 1, Y Axis Position Error States (Note: GPS Measurements Only)

----	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

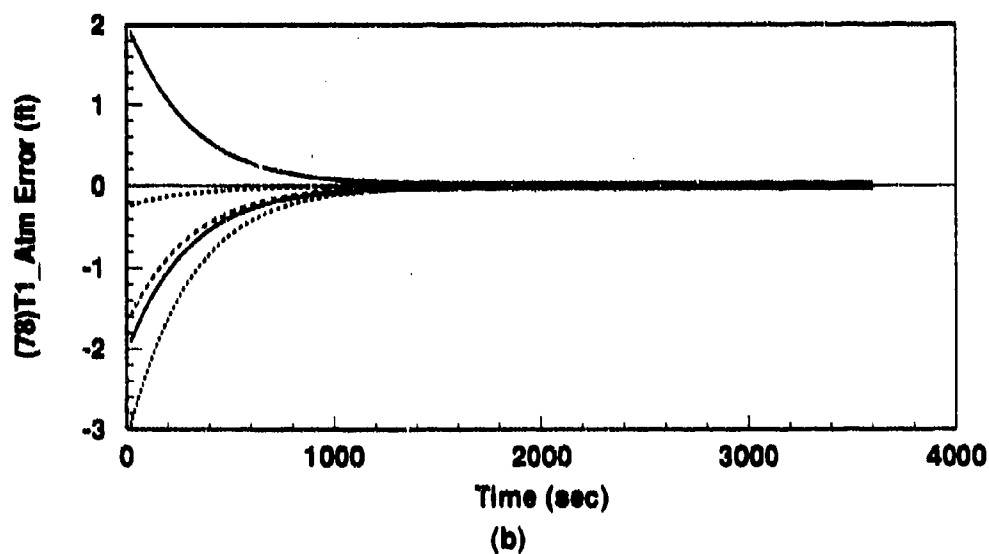
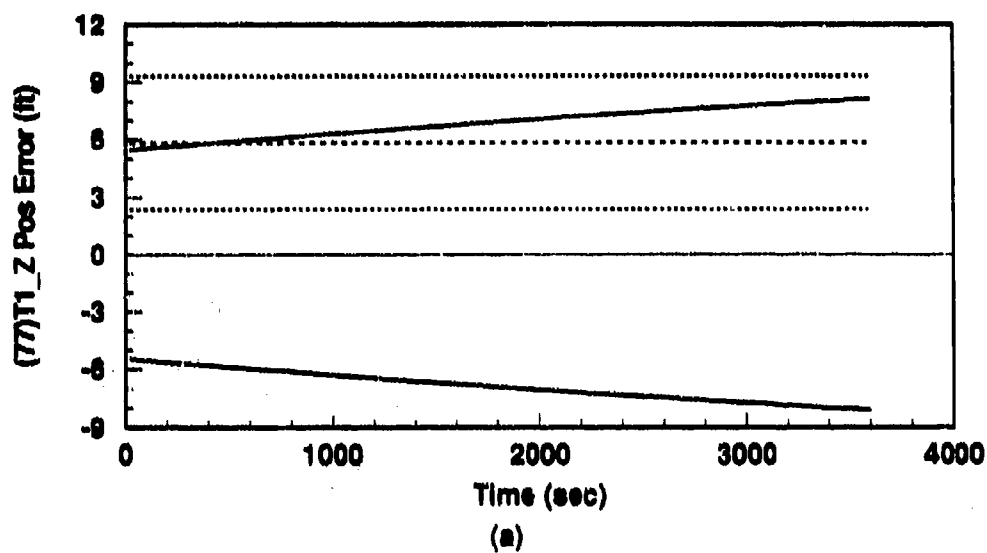


Figure H.20. Alignment: 128-State Model (a) Transponder 1, Z Axis Position (b) Transponder 1, Atmospheric Propagation Error States (Note: GPS Measurements Only)

- - - -	Mean Error = $\bar{M}_e - (M_e)_{true}$
.....	Mean Error $\pm \sigma_{true}$
— — —	$0 \pm \sigma_{filter}$

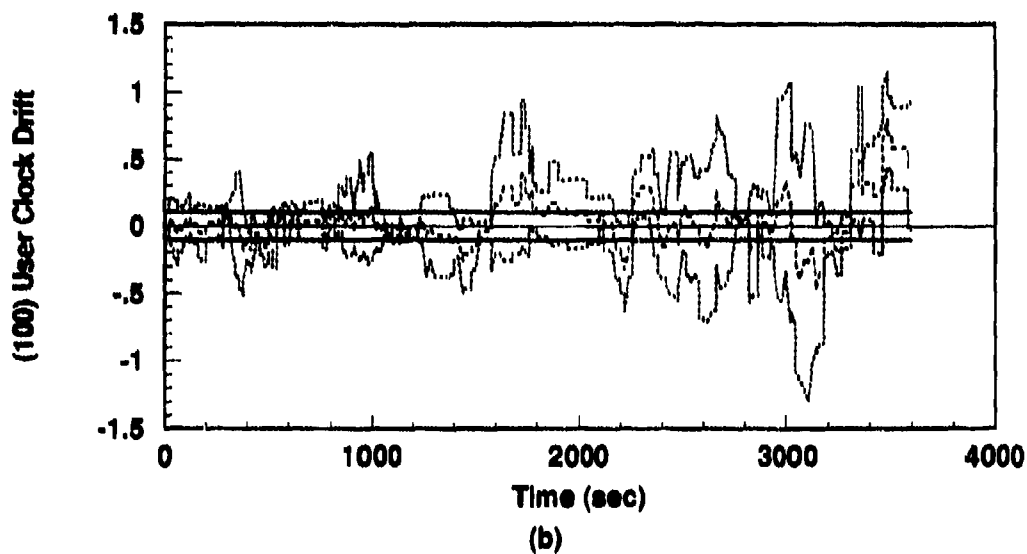
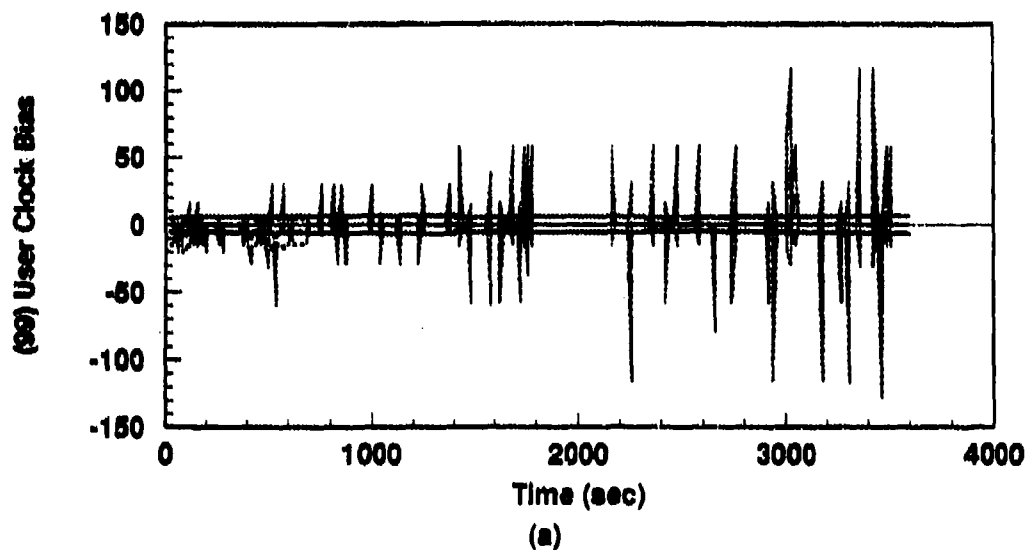


Figure H.21. Flight: 128-State Model (a) User Clk Bias and (b) User Clock Drift States
(Note: GPS Measurements Only)

----	Mean Error = $\bar{M}_t - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

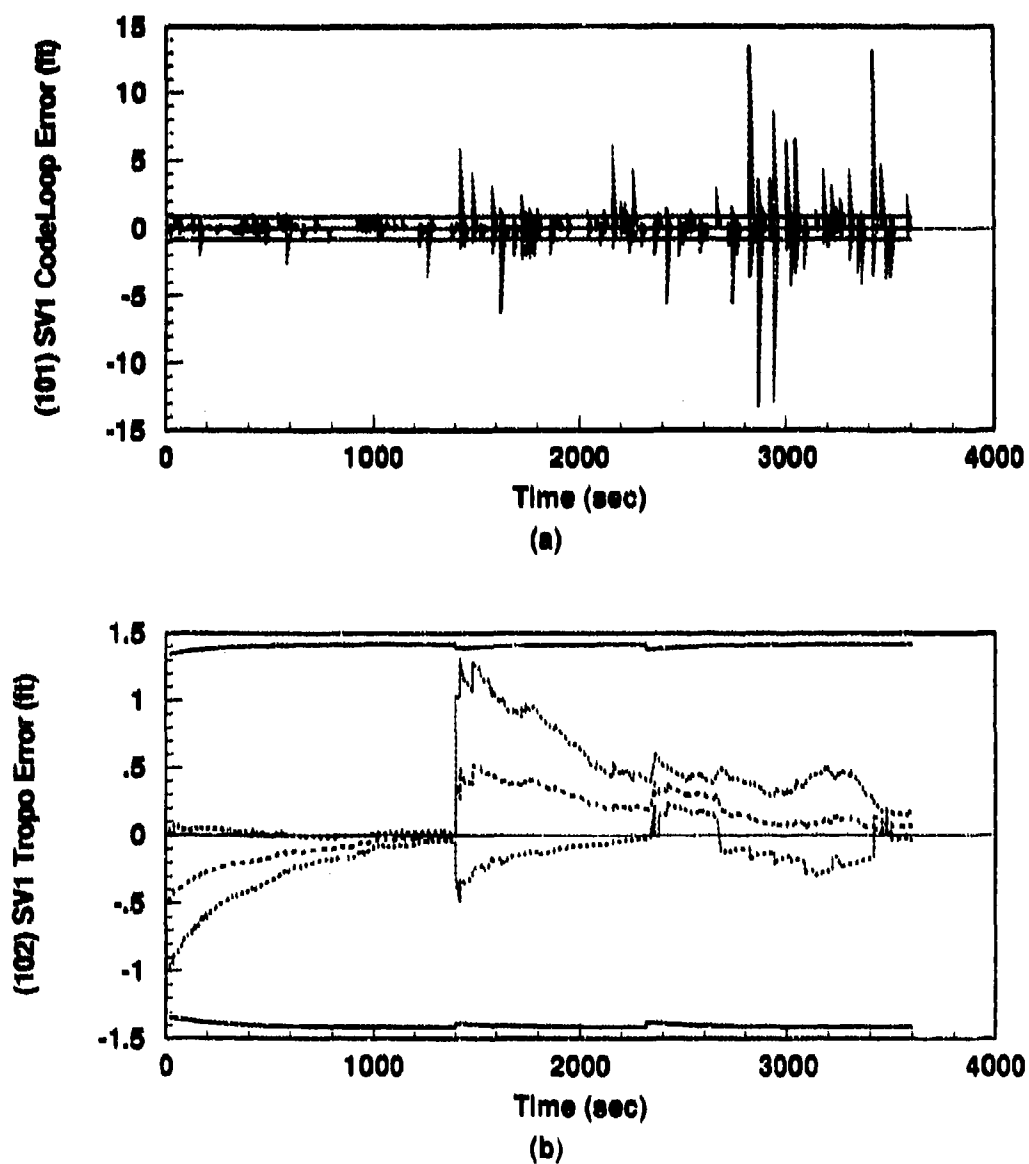


Figure H.22. Flight: 128-State Model (a) SV 1, CodeLoop and (b) SV 1, Atmospheric Propagation Error States (Note: GPS Measurements Only)

----	Mean Error = $\bar{M}_x - (M_x)_{true}$
-----	Mean Error $\pm \sigma_{true}$
———	$0 \pm \sigma_{filter}$

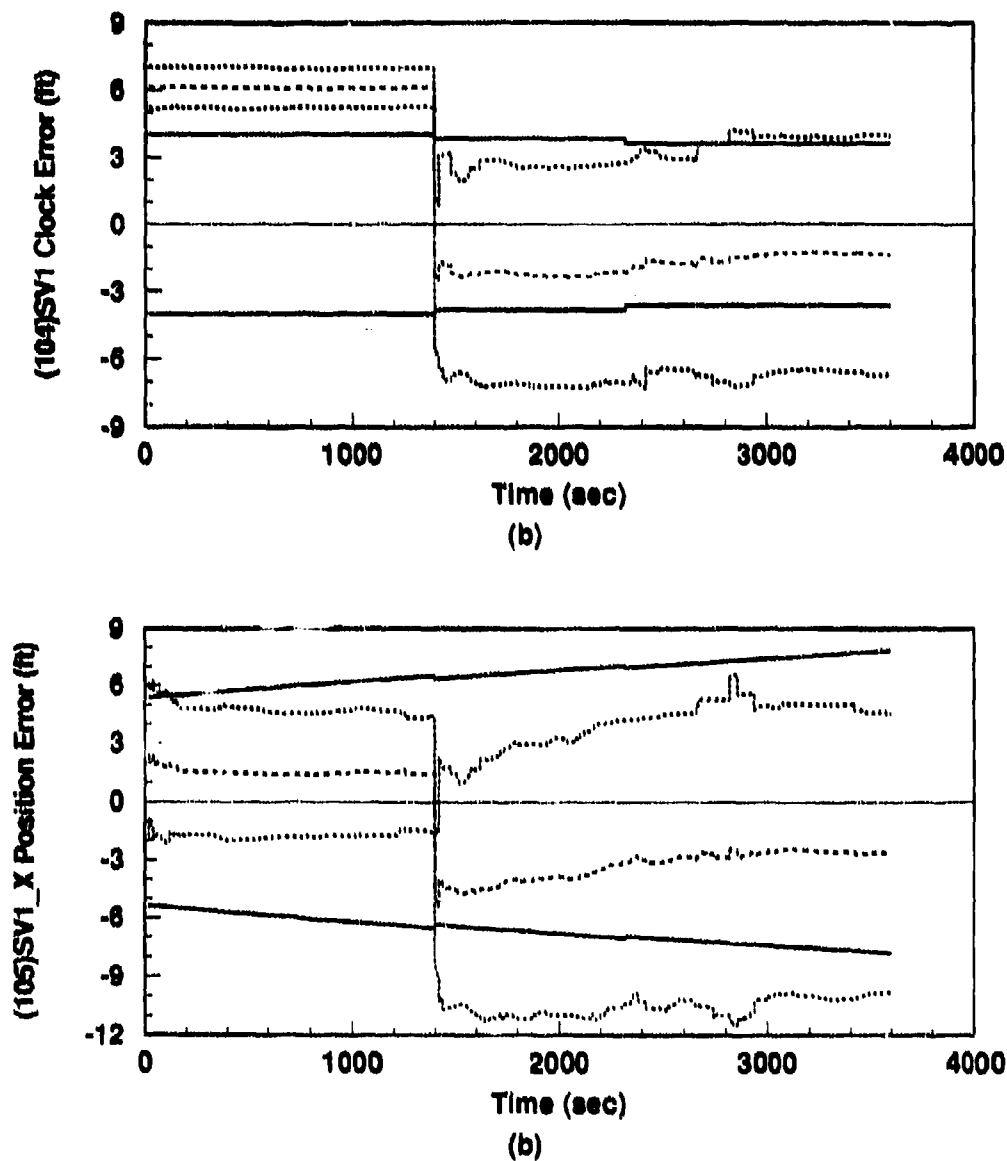


Figure H.23. Flight: 128-State Model (a) SV 1, Clock and (b) SV 1, X Position Error States (Note: GPS Measurements Only)

.....	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
----	$0 \pm \sigma_{filter}$

Appendix I. 128-State INS-RRS-GPS Alignment and Flight Simulations

This appendix contains the plots from a 10-run Monte Carlo alignment and a 10-run Monte Carlo flight simulation of the 128-state error model which incorporates the truncated LN-93 INS, the revised baro-altimeter model, RRS, and GPS. Initial conditions are once again chosen for an alignment at Holloman AFB, NM. These results (using GPS measurements) are compared to results from simulations in Appendix F, Appendix G, and Appendix H.

Variables plotted in this appendix are defined exactly in the same manner as those in Appendix C; all statistical quantities are calculated in the manner described at the beginning of Appendix C.

I.1 128-State Model: Alignment Using Holloman Initial Conditions

The plots in this section represent results of a 10-run Monte Carlo alignment simulation. In these alignments, INS aiding consists of velocity measurements, baro-altimeter measurements, RRS transponder range measurements, and GPS pseudo-range measurements.

The purpose of this set of runs is to establish that the software function for the INS-RRS-GPS combination is comparable to (or better than) that achieved in the Litton flight runs reported in Appendices C through F. In addition, results are expected to be somewhat better than those reported in Appendix G and Appendix H.

Note that the plot for state 13, ΔS_4 , is zero for all time during all alignment simulations. This is a normal condition due to the variable's dependence on altitude rate. The state becomes non-zero during flight runs. The GPS User clock states exhibit large magnitude transients during the initial phase of the alignment runs. As a result, an additional "window" is plotted on the page which follows the first user clock plots. The purpose is to demonstrate the steady state behavior of these critical error states. All other plots are discussed in Chapter VI (Results).

The filter computed error estimates $[\sigma_{filter}]$ are also compared to similar plots contained in the Litton reference documentation [20]. Comparisons are qualitative only and

are intended to demonstrate trends. Such comparisons are contained in Chapter VI (Results).

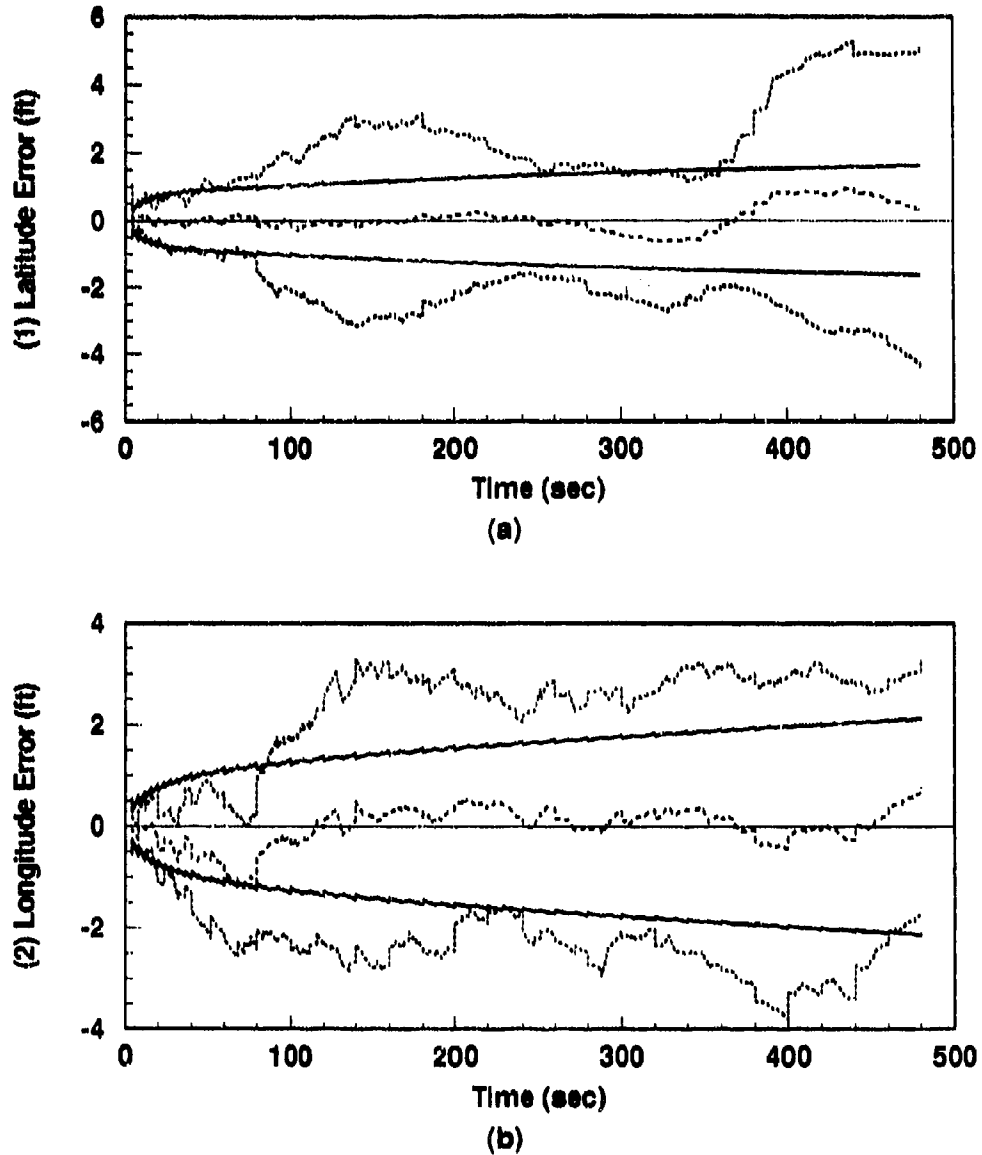


Figure I.1. Alignment: 128-State Model (a) Latitude and (b) Longitude Errors. (Note: Both GPS and RRS Measurements)

----	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{trim}$
-----	$0 \pm \sigma_{filter}$

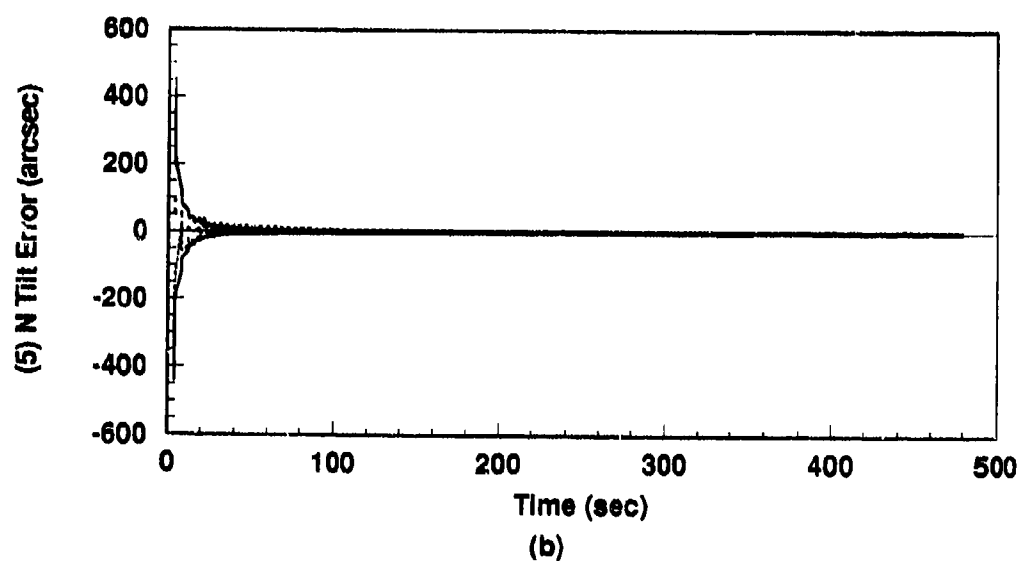
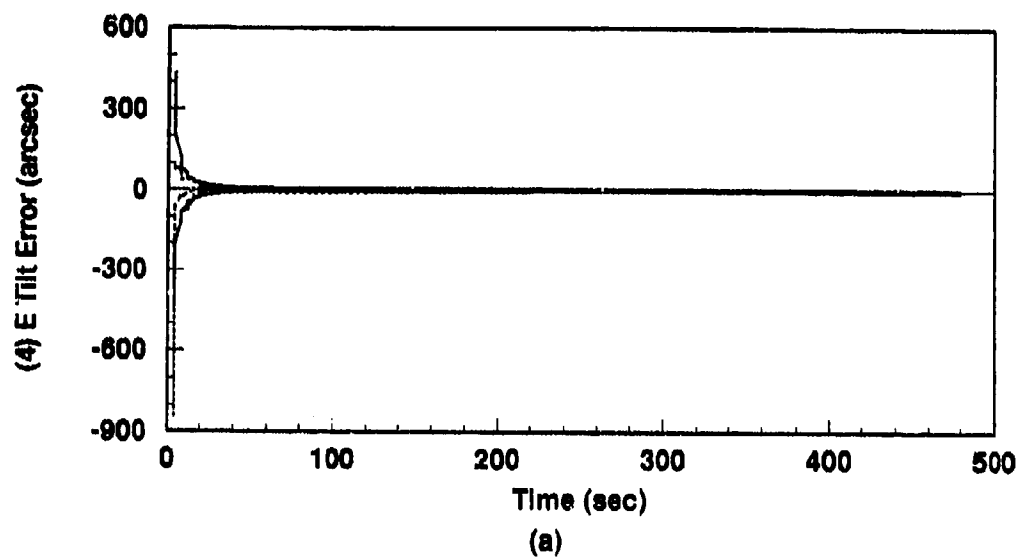


Figure I.2. Alignment: 128-State Model (a) East Tilt and (b) North Tilt Error States.
(Note: Both GPS and RRS Measurements)

----	$Mean\ Error = \hat{M}_x - (M_x)_{true}$
.....	$Mean\ Error \pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

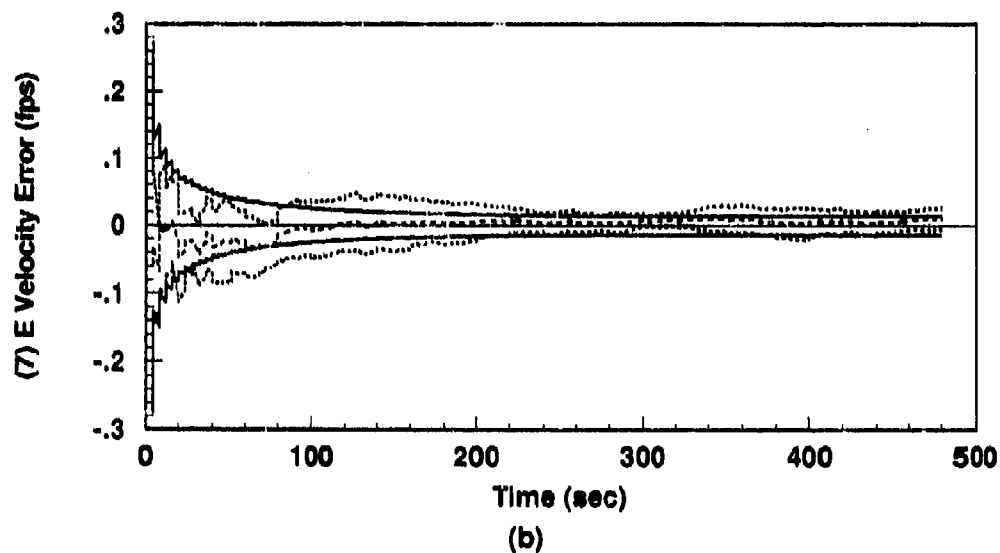
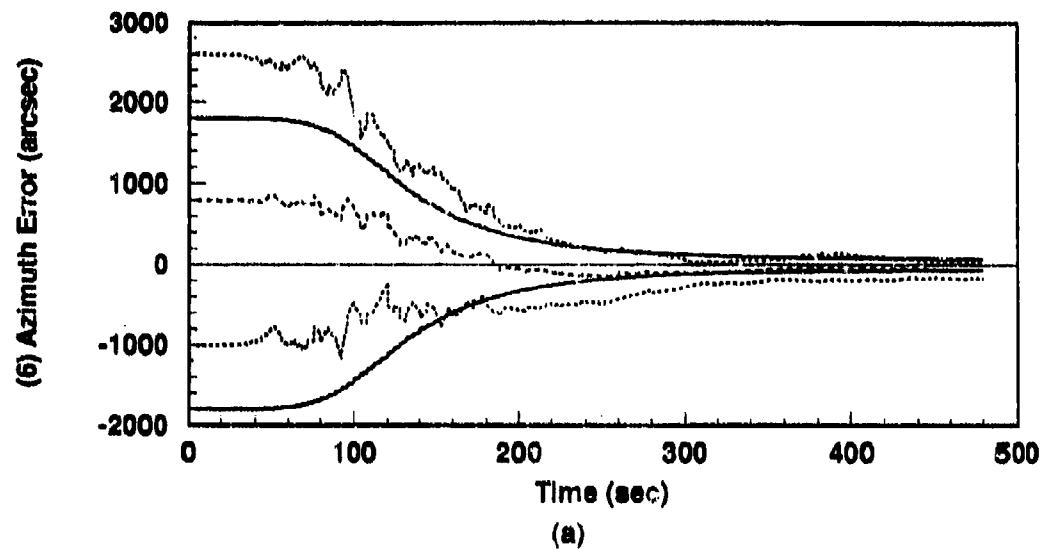


Figure I.3. Alignment: 128-State Model (a) Azimuth and (b) East Velocity Error States.
(Note: Both GPS and RRS Measurements)

----	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

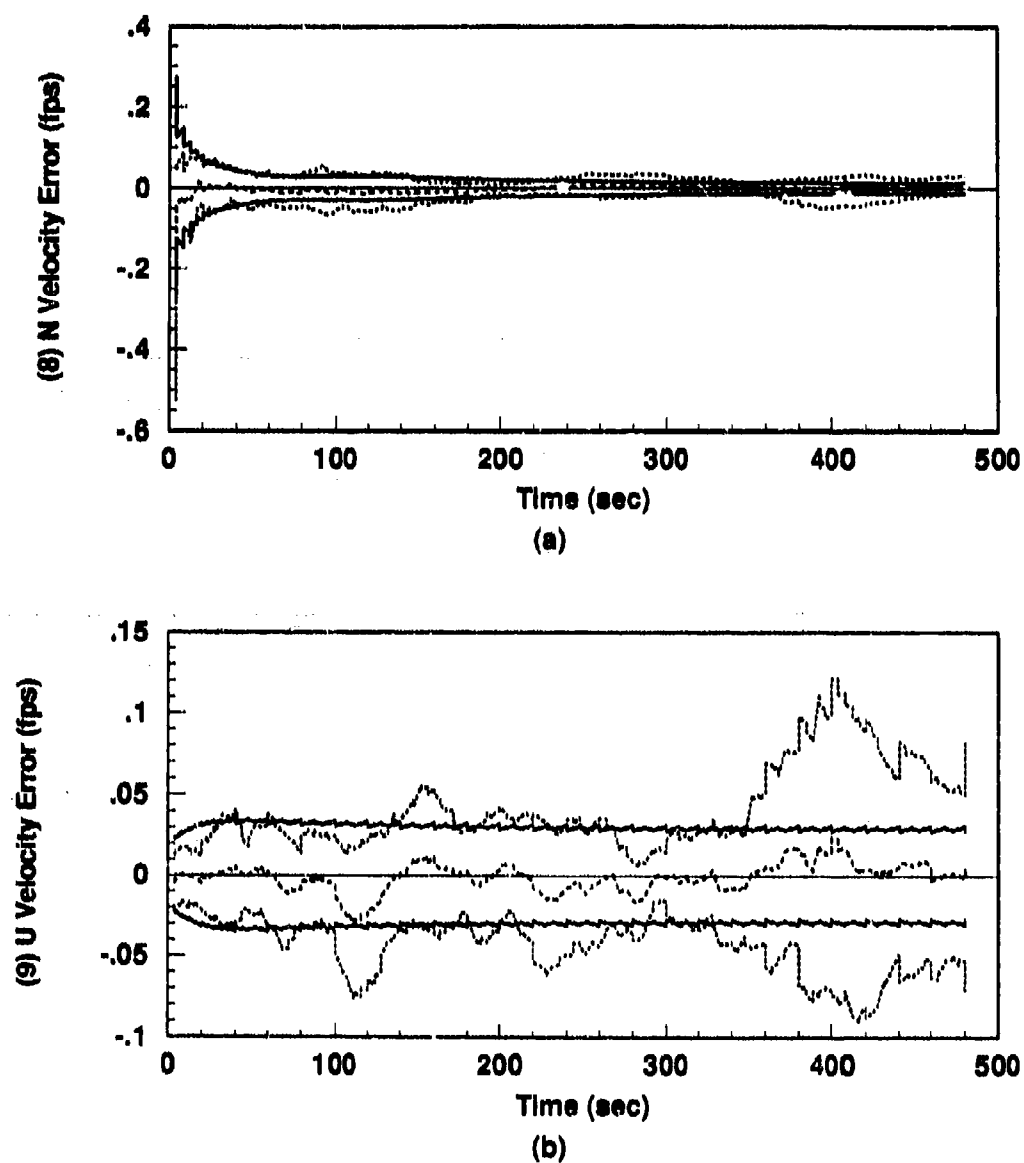


Figure I.4. Alignment: 128-State Model (a) North Velocity and (b) Vertical Velocity Error States. (Note: Both GPS and RRS measurements used.)

----	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
-----	$0 \pm \sigma_{filter}$

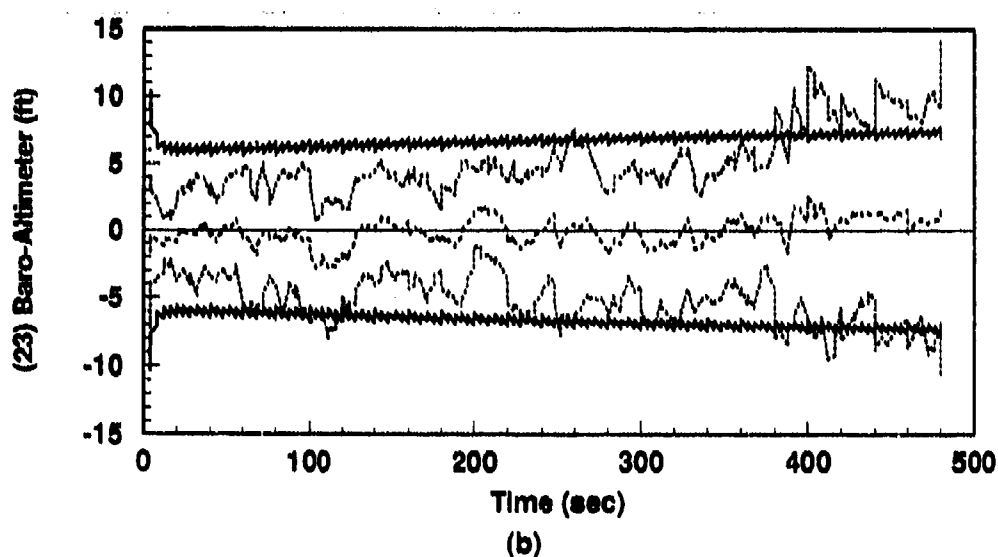
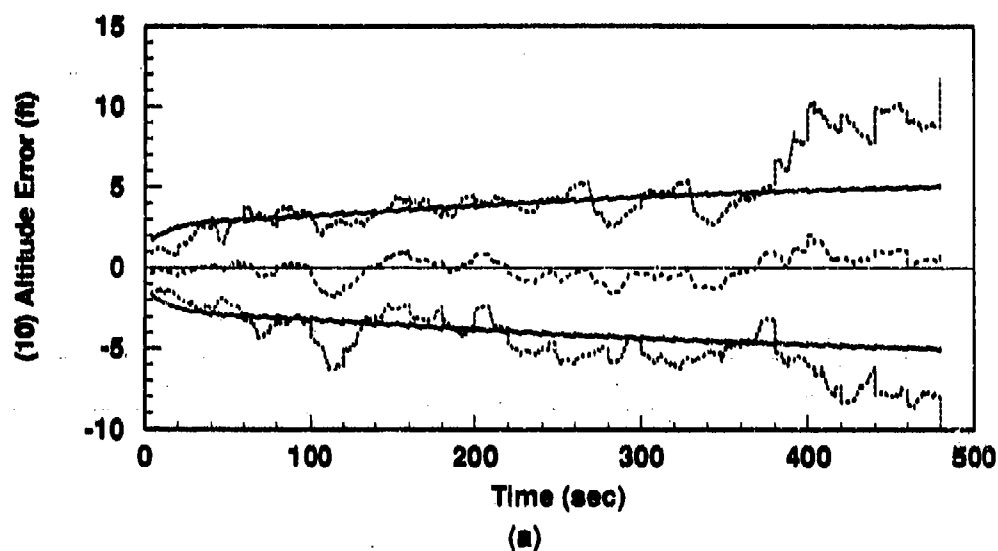


Figure I.5. Alignment: 128-State Model (a) INS Altitude Error State and (b) Baro-Altimeter Total Error. (Note: Both GPS and RRS Measurements)

----	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

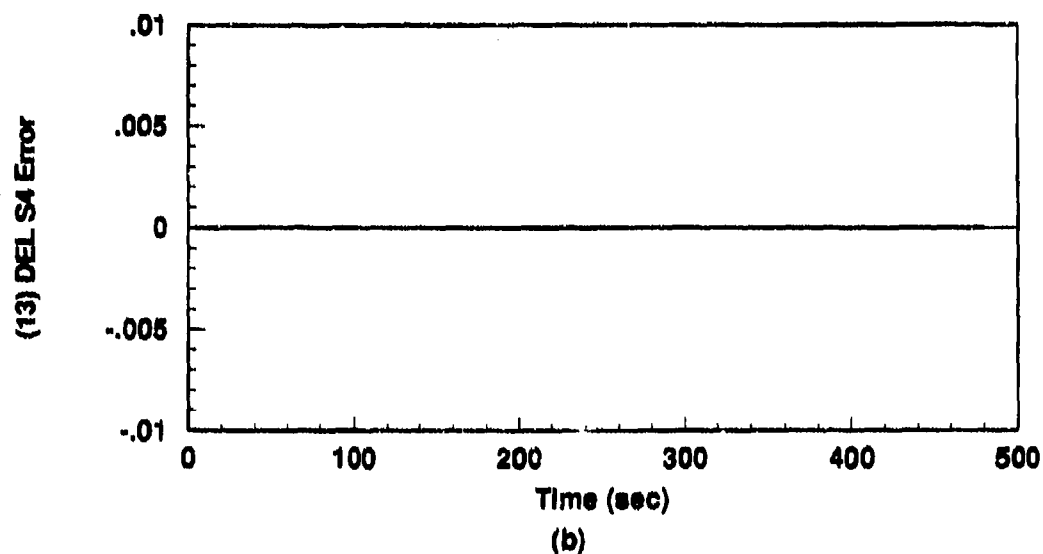
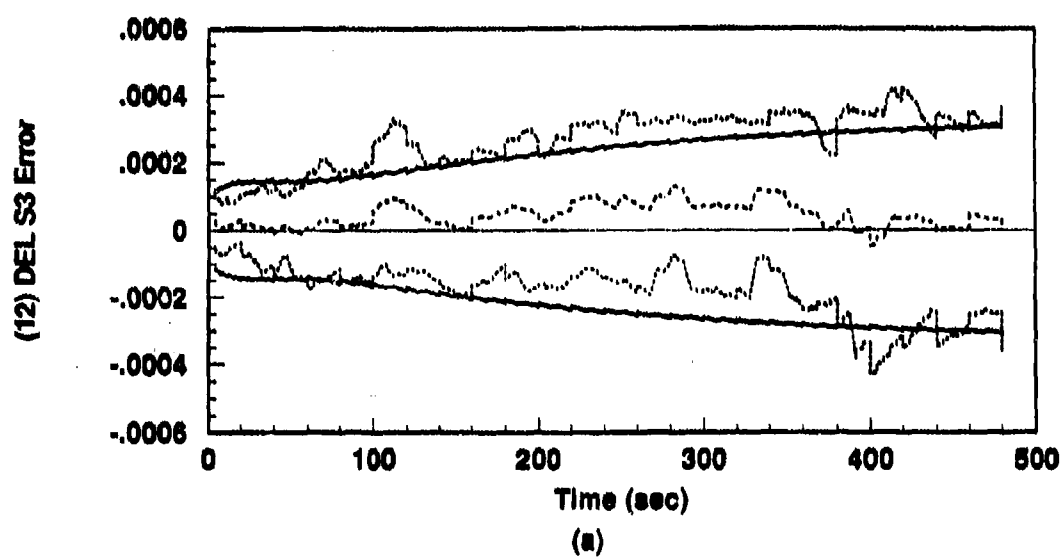


Figure I.6. Alignment: 128-State Model (a) ΔS_3 and (b) ΔS_4 Vertical Channel Aiding Error States. (Note: Both GPS and RRS Measurements)

----	Mean Error = $\bar{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

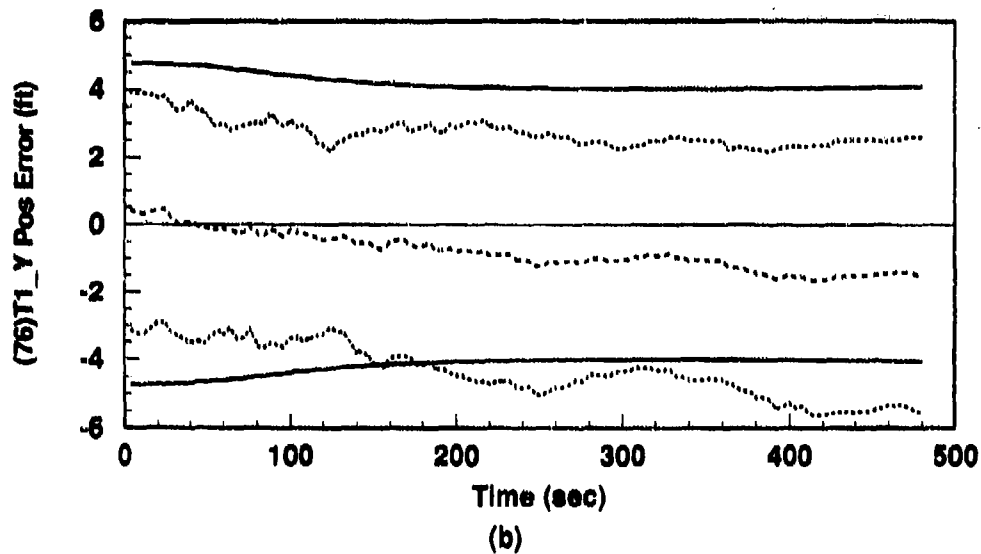
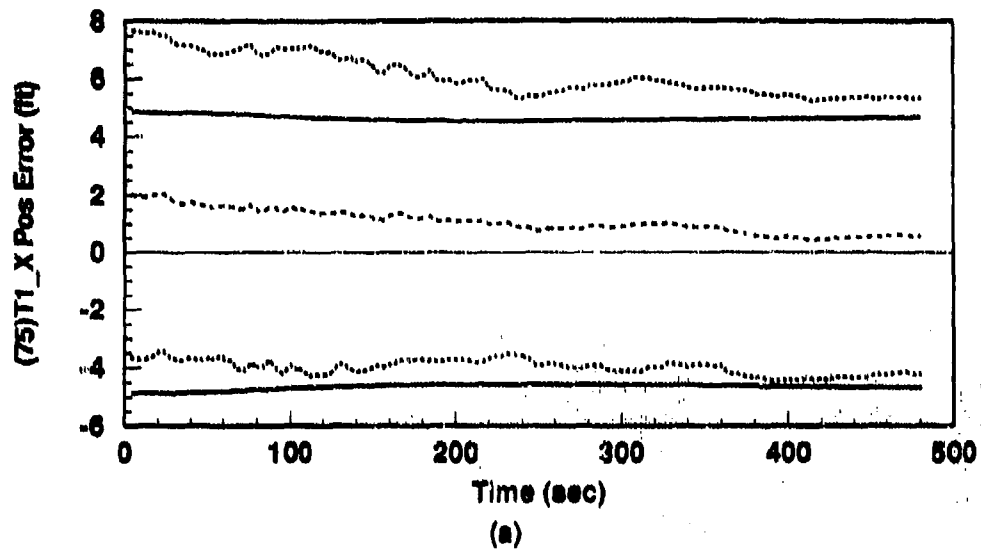


Figure I.7. Alignment: 128-State Model (a) Transponder 1, X Axis Position (b) Transponder 1, Y Axis Position Error States (Note: Both GPS and RRS Measurements)

.....	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

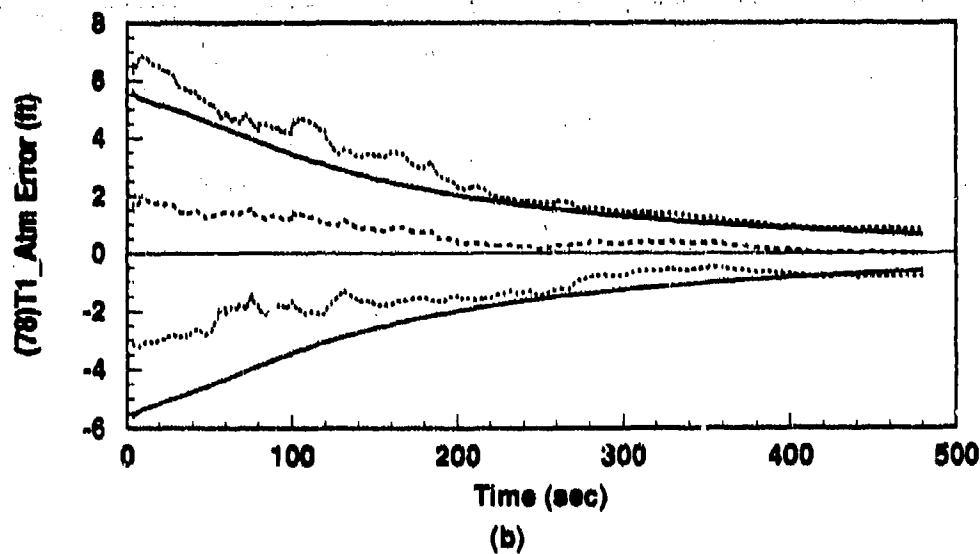
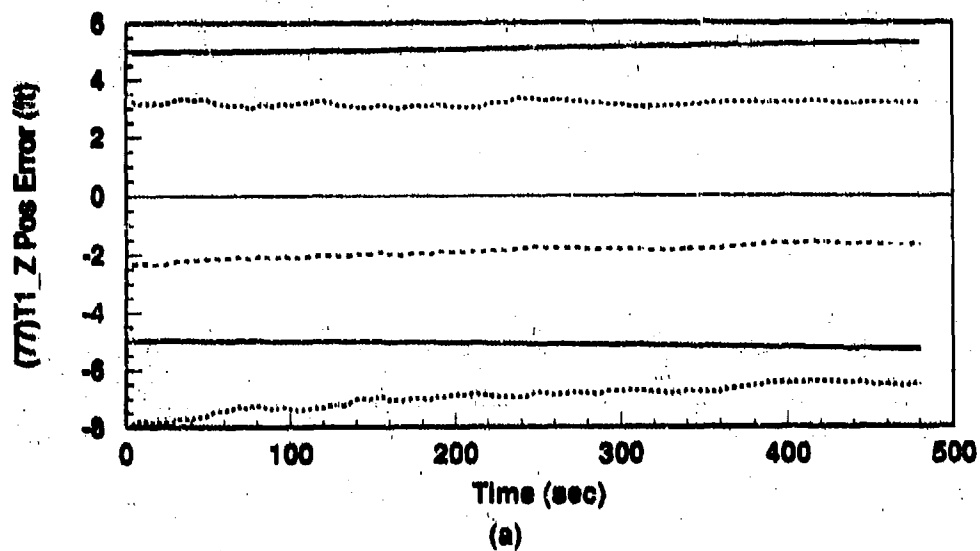


Figure I.8. Alignment: 128-State Model (a) Transponder 1, Z Axis Position (b) Transponder 1, Atmospheric Propagation Error States (Note: Both GPS and RRS Measurements)

----	Mean Error = $\bar{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
-----	$0 \pm \sigma_{filter}$

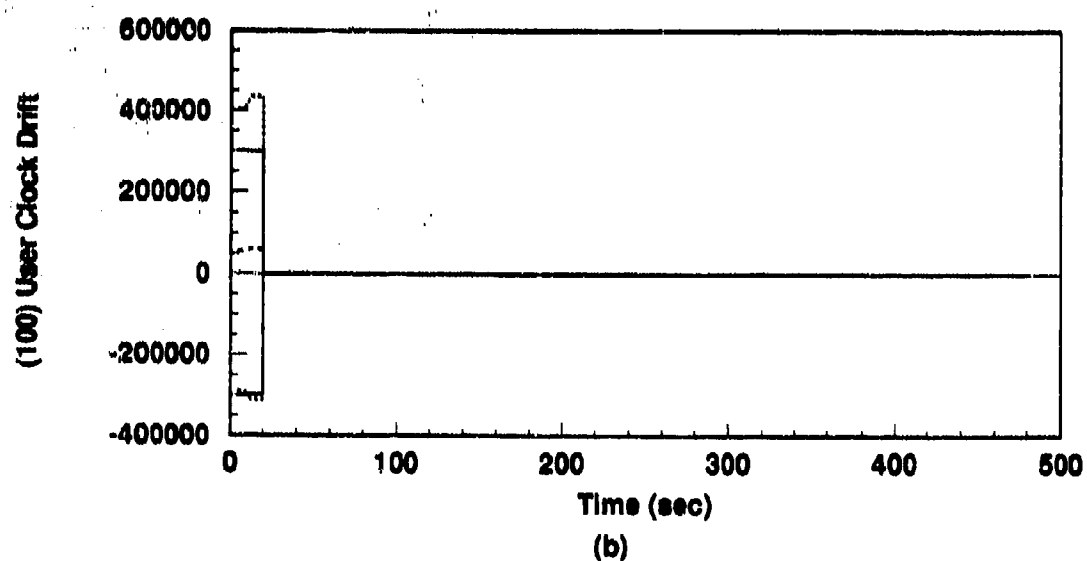
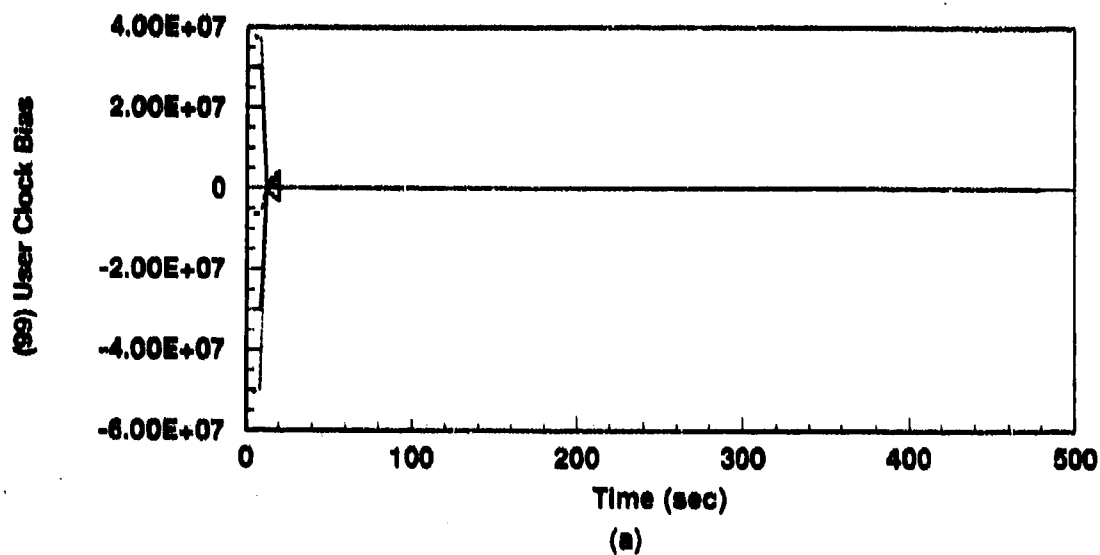


Figure I.9. Alignment: 128-State Model (a) User Clk Bias and (b) User Clock Drift States (Note: Both GPS Measurements Only)

----	$Mean\ Error = \hat{M}_x - (M_x)_{true}$
.....	$Mean\ Error \pm \sigma_{lim}$
—	$0 \pm \sigma_{filter}$

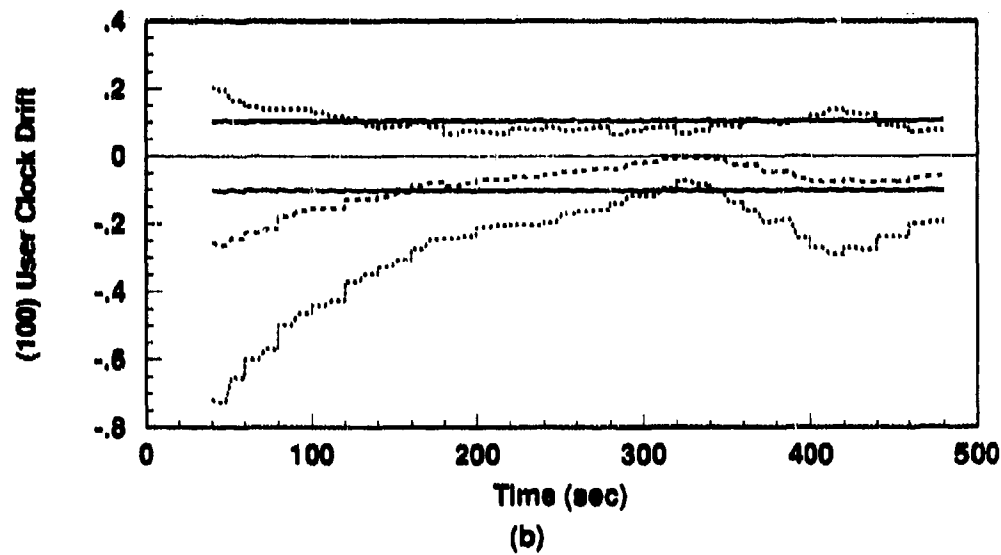
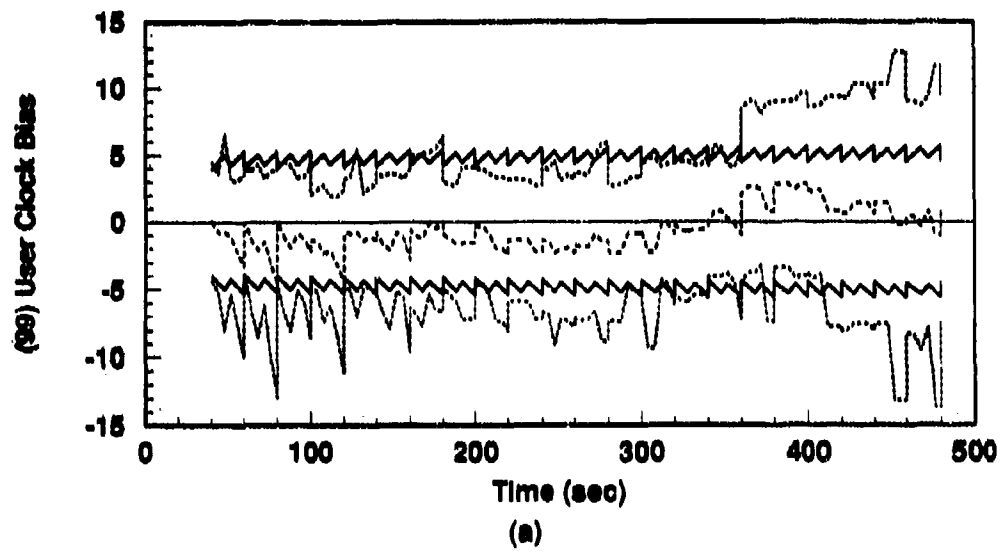


Figure I.10. Alignment: 128-State Model: *Steady State* (a) User Clk Bias and (b) User Clock Drift States (Note: GPS and RRS Measurements)

----	$Mean\ Error = \bar{M}_x - (M_x)_{true}$
.....	$Mean\ Error \pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

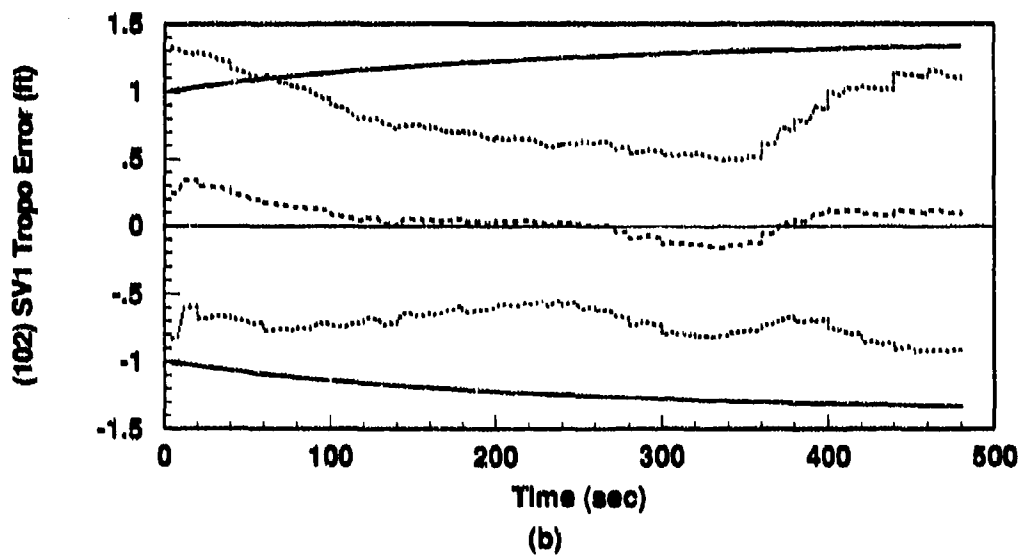
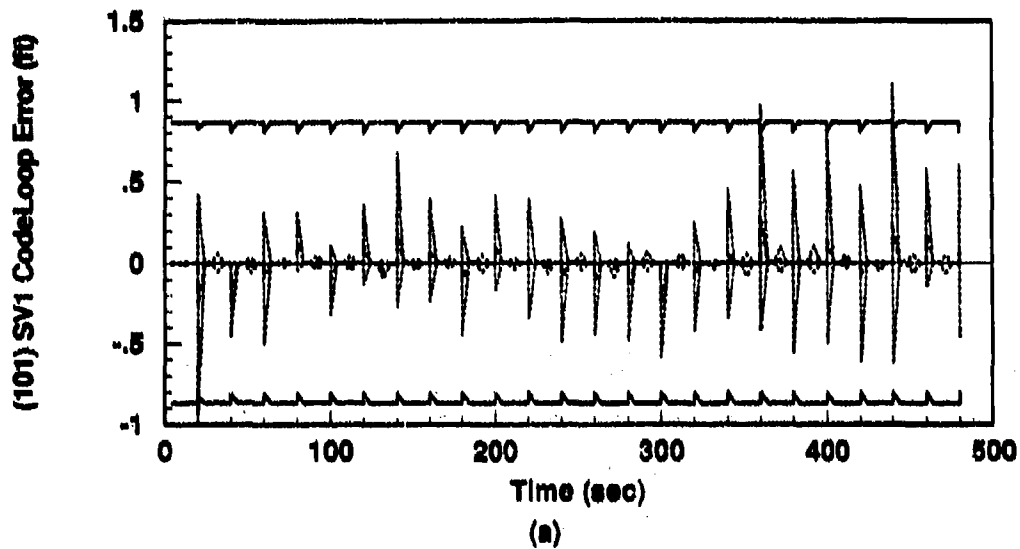


Figure I.11. Alignment: 128-State Model (a) SV 1, CodeLoop and (b) SV 1, Atmospheric Propagation Error States (Note: Both GPS Measurements Only)

----	Mean Error = $\bar{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

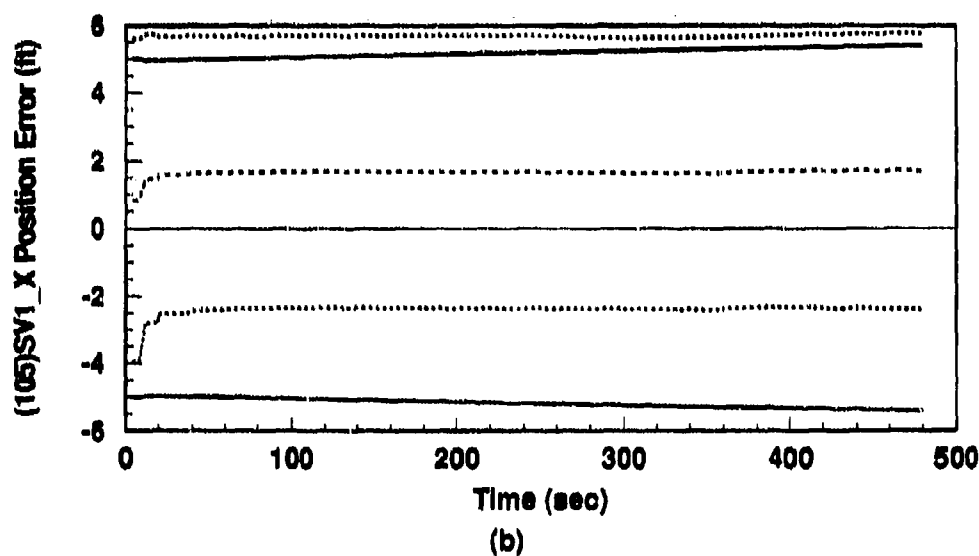
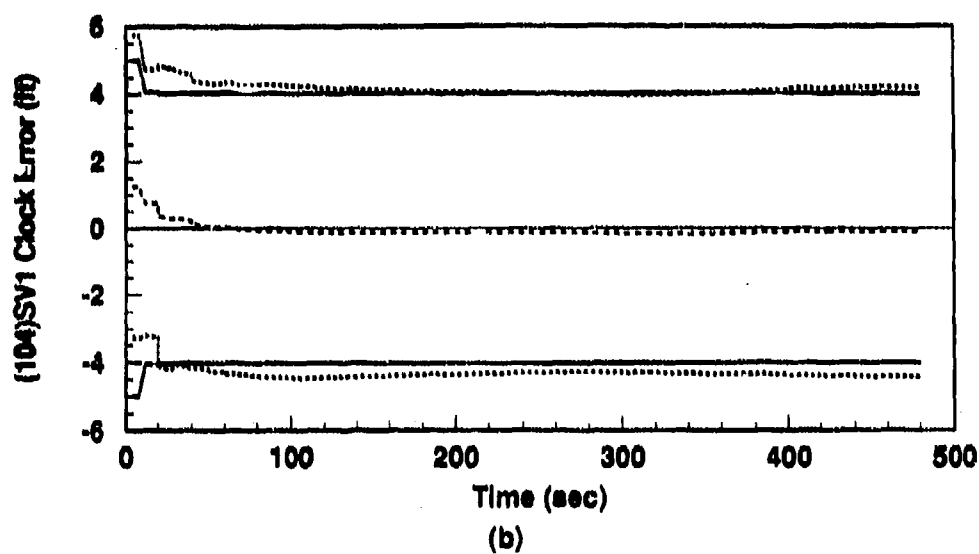


Figure I.12. Alignment: 128-State Model (a) SV 1, Clock and (b) SV 1, X Position Error States (Note: Both GPS Measurements Only)

.....	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

1.2 128-State Model: Fighter Flight Using Holloman Initial Conditions

The plots in this section represent results of a 10-run flight simulation in which a flight profile (as described in Chapter III) is used to characterize the performance of the LN-93 with RRS aiding for a fighter mission originating and terminating at Holloman AFB, NM.

The purpose of this set of runs is to establish that the software function for the INS-RRS-GPS combination is comparable to (or better than) that achieved in the Litton flight runs reported in Appendices C through F. In addition, results are expected to be somewhat better than those reported in Appendix G.

The filter computed error estimates [σ_{filter}] are also compared to similar plots contained in the Litton reference documentation [20]. Comparisons are qualitative only and are intended to demonstrate trends. Such comparisons are contained in Chapter VI (Results).

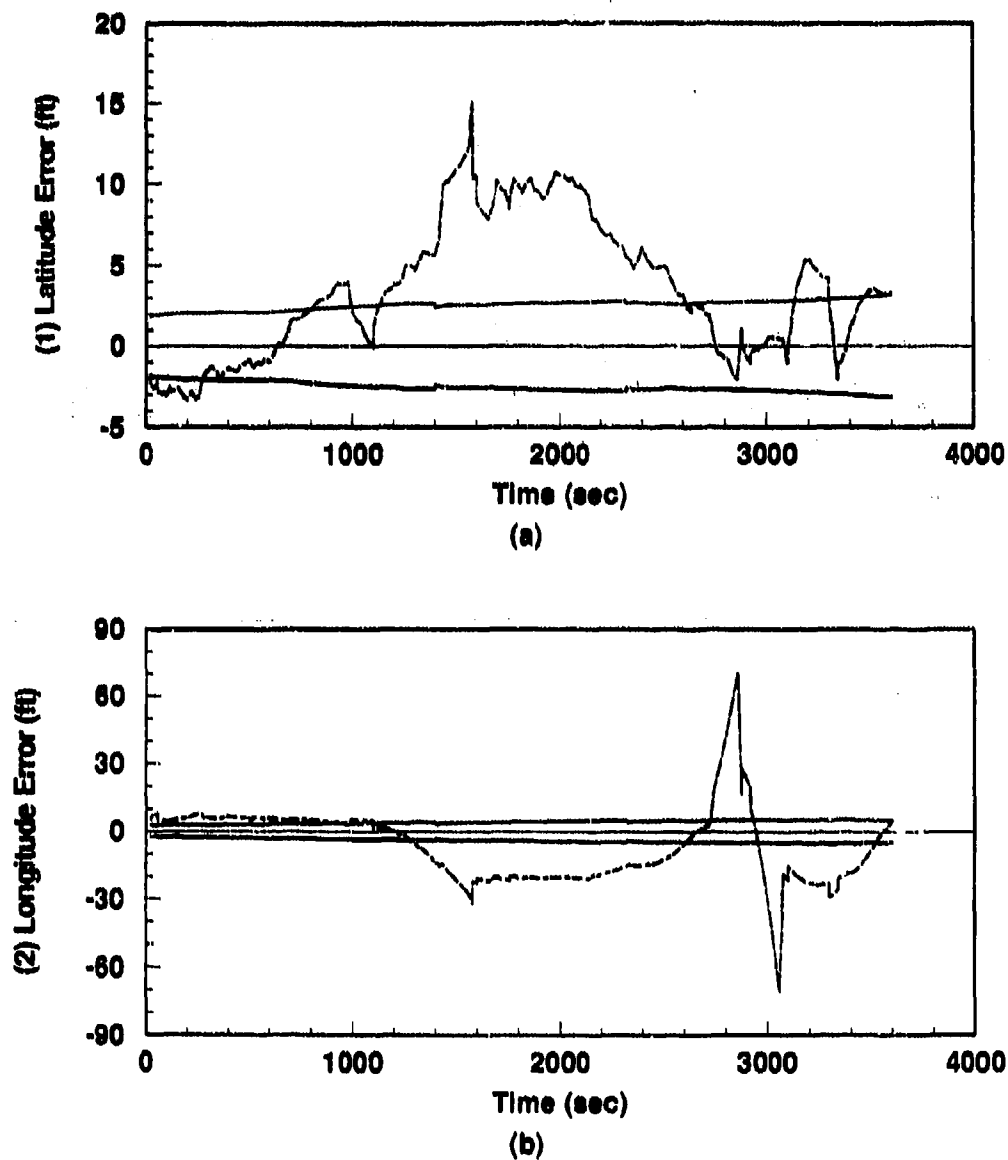


Figure I.13. Flight: 128-State Model (a) Latitude and (b) Longitude Error States. (Note: Both GPS and RRS Measurements)

----	Mean Error = $\bar{M}_r - (M_r)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

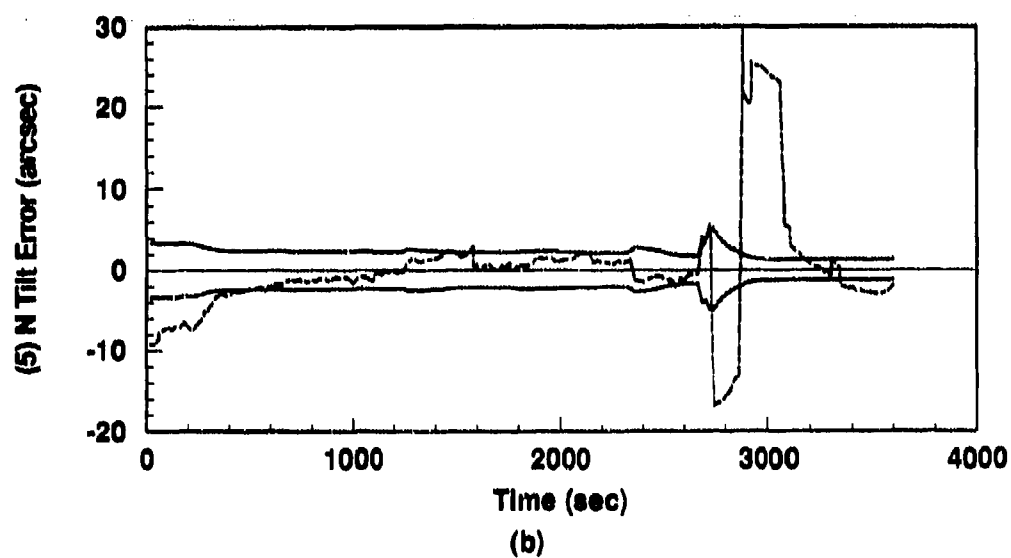
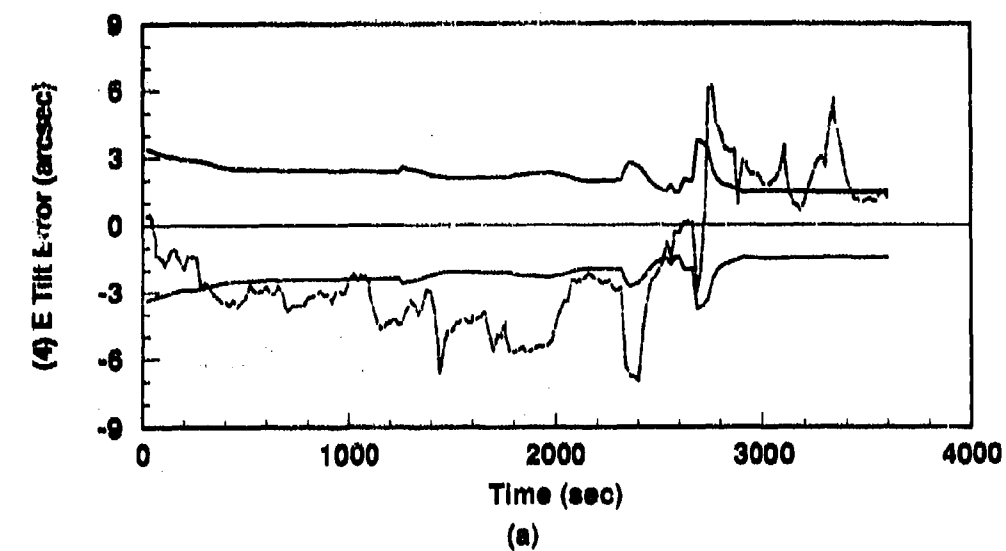


Figure I.14. Flight: 128-State Model (a) East Tilt and (b) North Tilt Error States.
(Note: Both GPS and RRS Measurements)

----	$Mean\ Error = \bar{M}_r - (M_r)_{true}$
.....	$Mean\ Error \pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

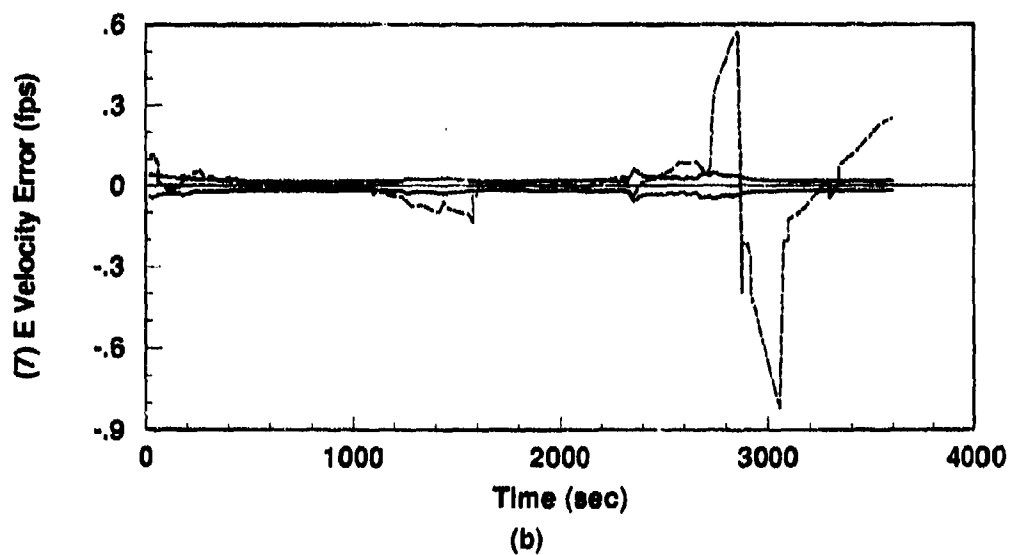
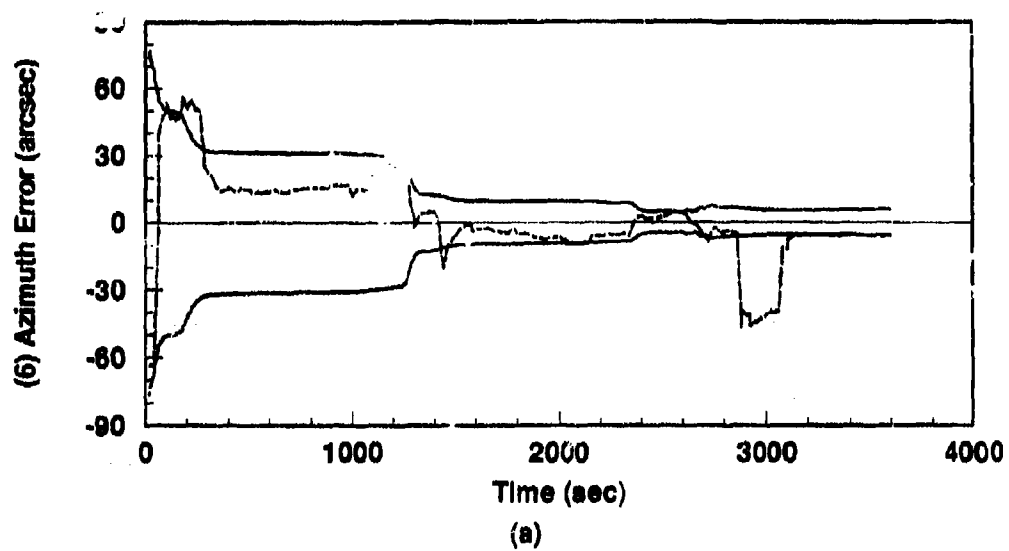
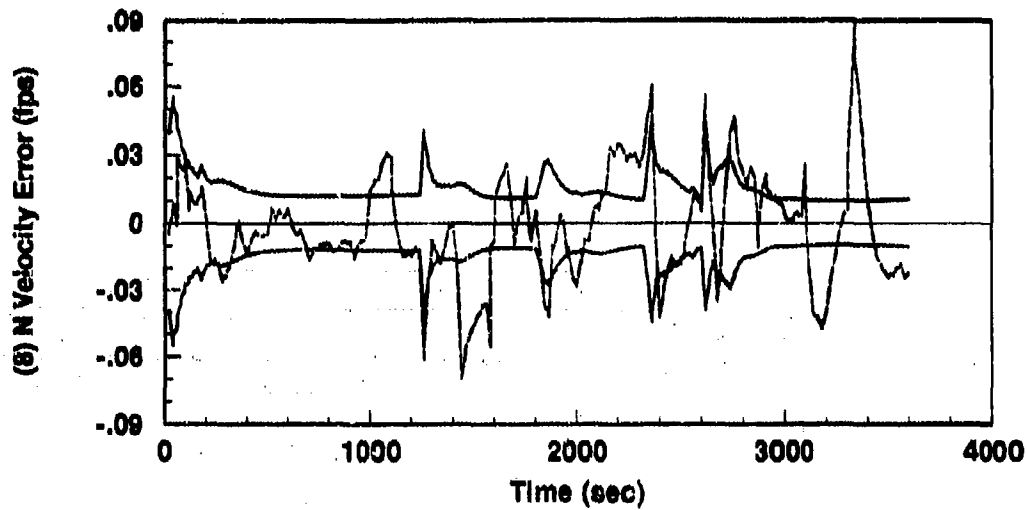
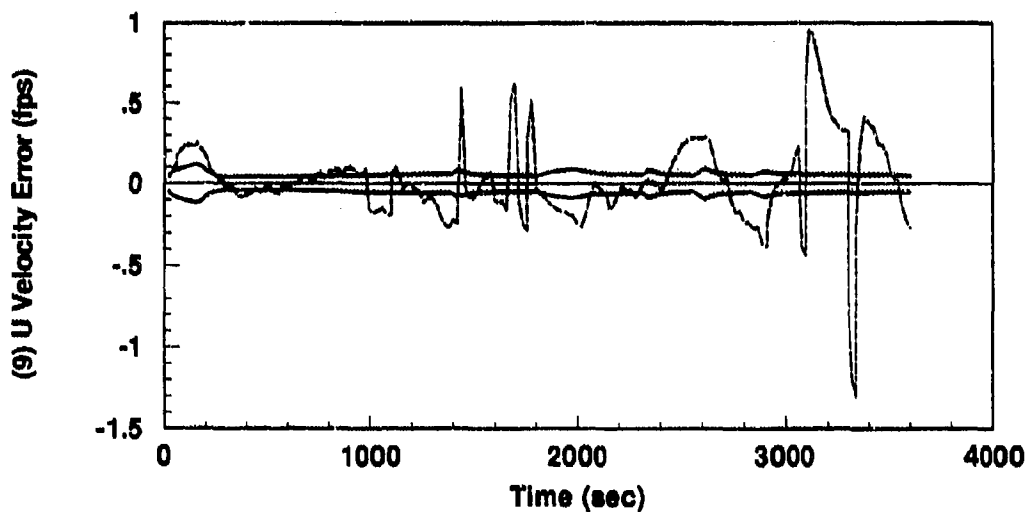


Figure I.15. Flight: 128-State Model (a) Azimuth and (b) East Velocity Error States.
(Note: Both GPS and RRS Measurements)

----	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$



(a)



(b)

Figure I.16. Flight: 128-State Model (a) North Velocity and (b) Vertical Velocity Error States. (Note: Both GPS and RRS Measurements)

----	$Mean\ Error = \hat{M}_x - (M_x)_{true}$
.....	$Mean\ Error \pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

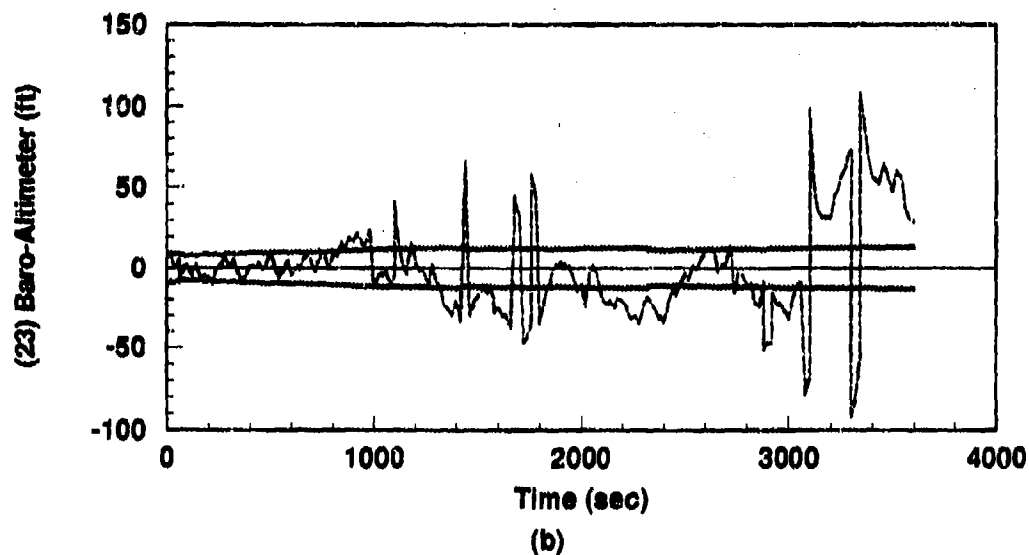
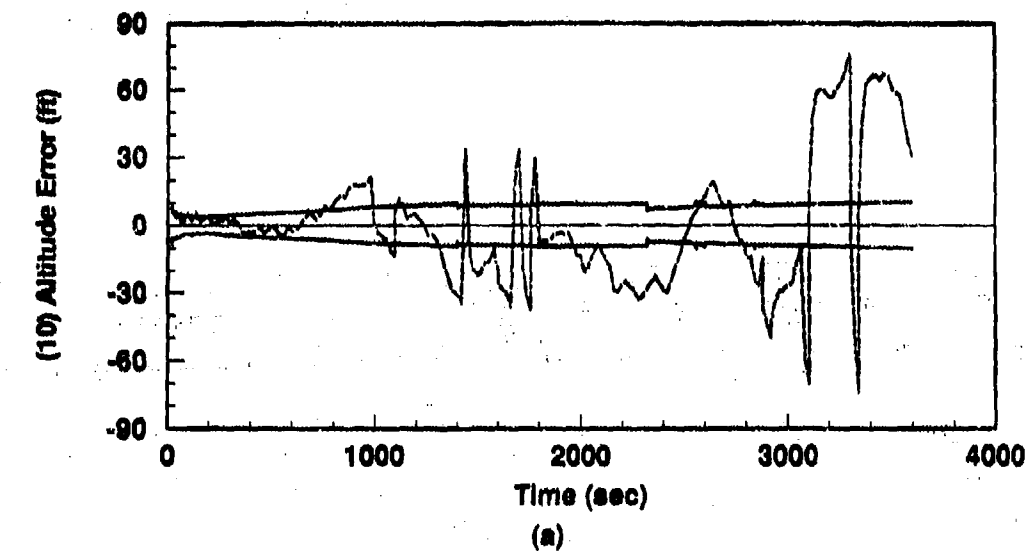
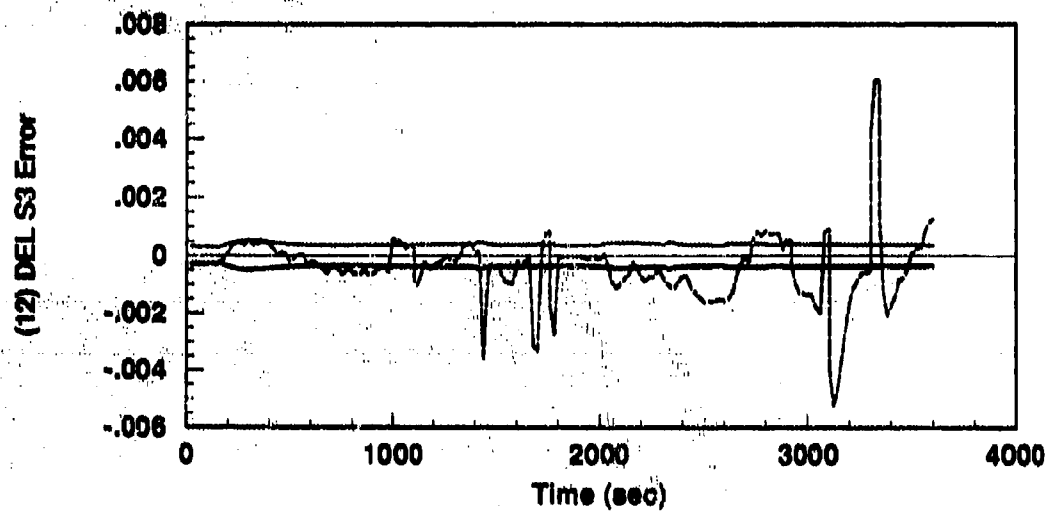
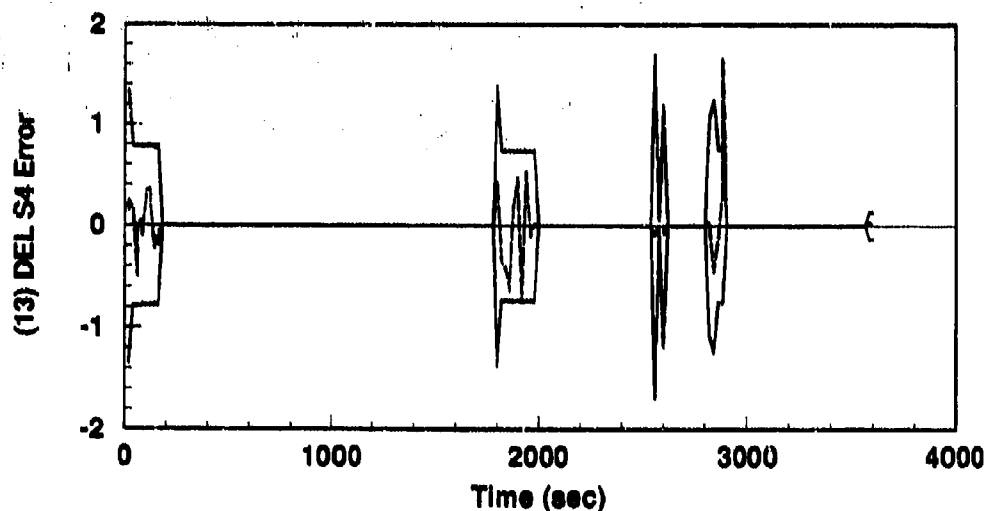


Figure I.17. Flight: 128-State Model (a) INS Altitude Error State and (b) Baro-Altimeter Total Error. (Note: Both GPS and RRS Measurements)

----	Mean Error = $\hat{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$



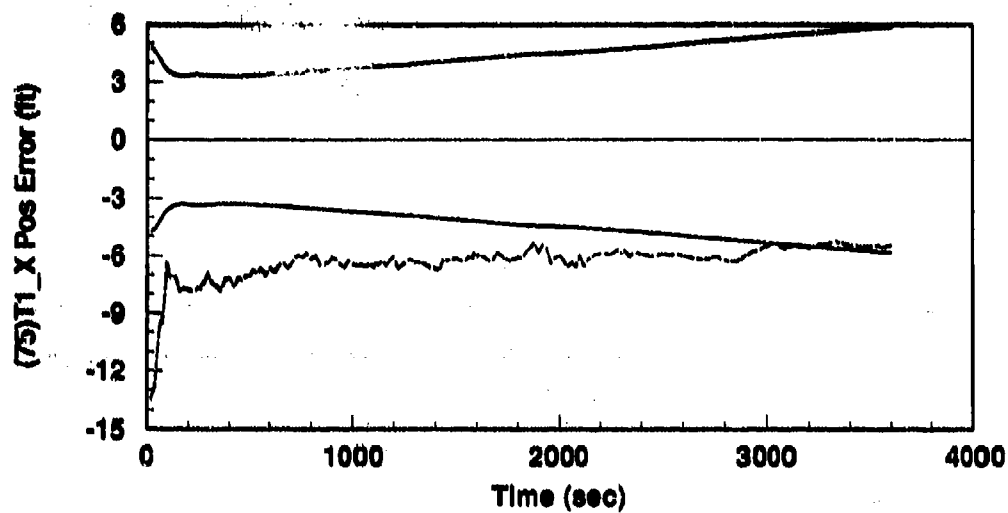
(a)



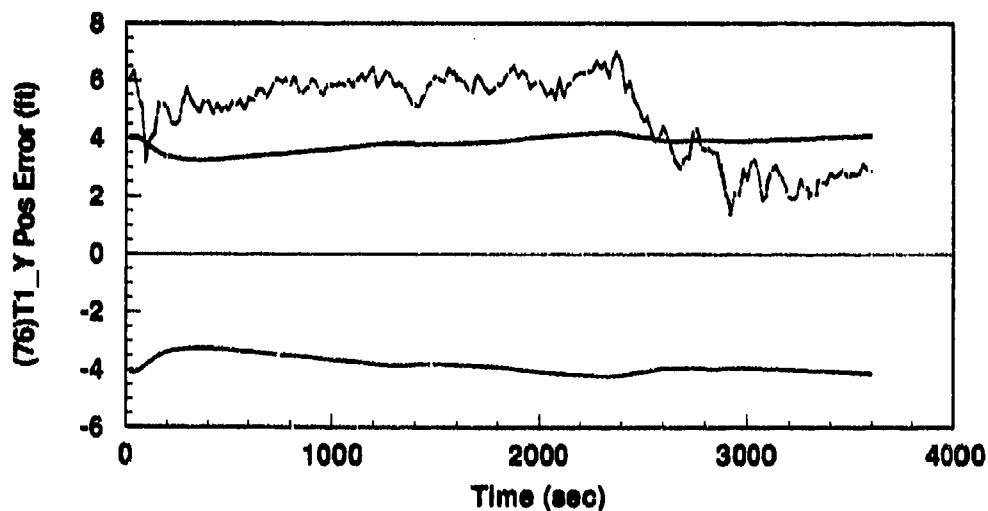
(b)

Figure I.18. Flight: 128-State Model (a) ΔS_3 and (b) ΔS_4 Vertical Channel Aiding Error States. (Note: Both GPS and RRS Measurements)

----	Mean Error = $\bar{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$



(a)



(b)

Figure I.19. Alignment: 128-State Model (a) Transponder 1, X Axis Position (b) Transponder 1, Y Axis Position Error States (Note: Both GPS and RRS Measurements)

----	Mean Error = $\bar{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

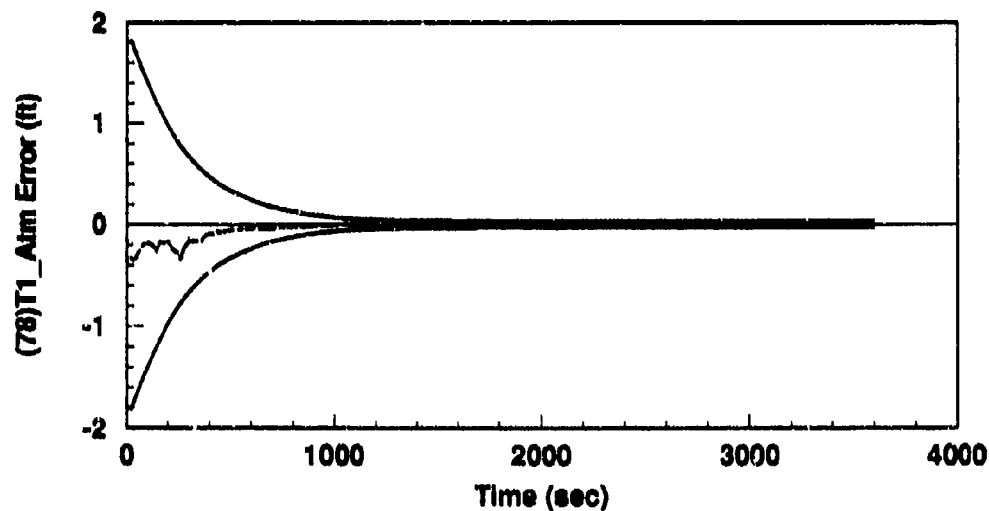
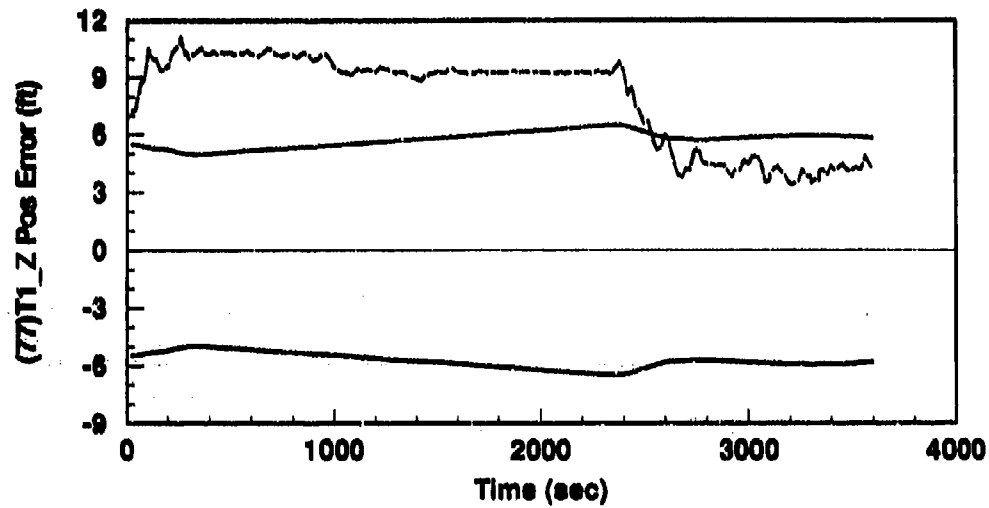
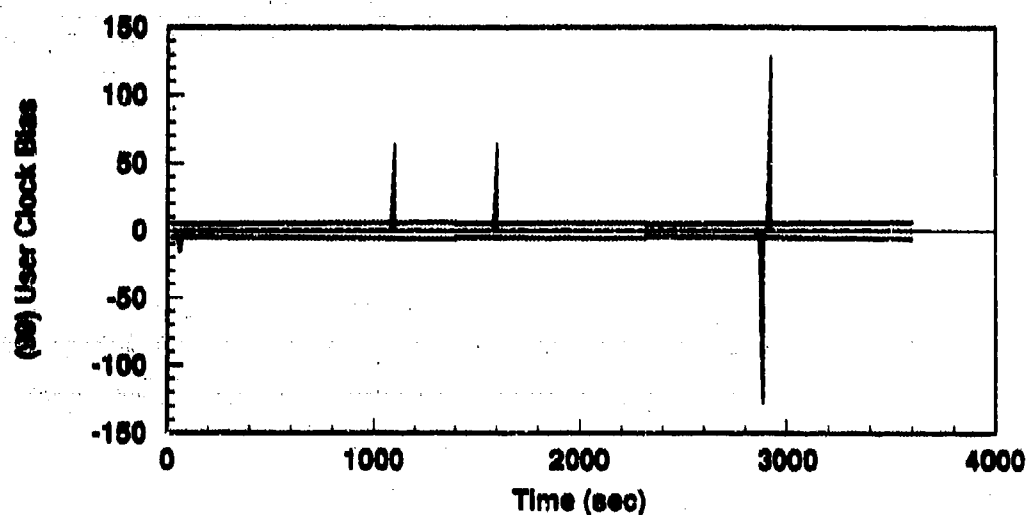
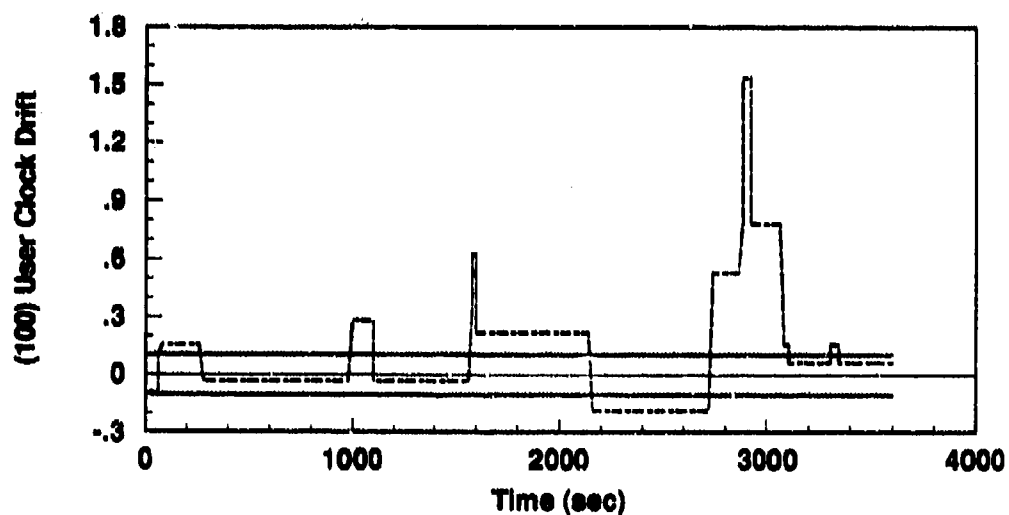


Figure I.20. Alignment: 128-State Model (a) Transponder 1, Z Axis Position (b) Transponder 1, Atmospheric Propagation Error States (Note: Both GPS and RRS Measurements)

----	$Mean\ Error = \bar{M}_x - (M_x)_{true}$
.....	$Mean\ Error \pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$



(a)



(b)

Figure I.21. Alignment: 128-State Model (a) User Clk Bias and (b) User Clock Drift States (Note: RRS and Both GPS Measurements)

----	Mean Error = $\bar{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

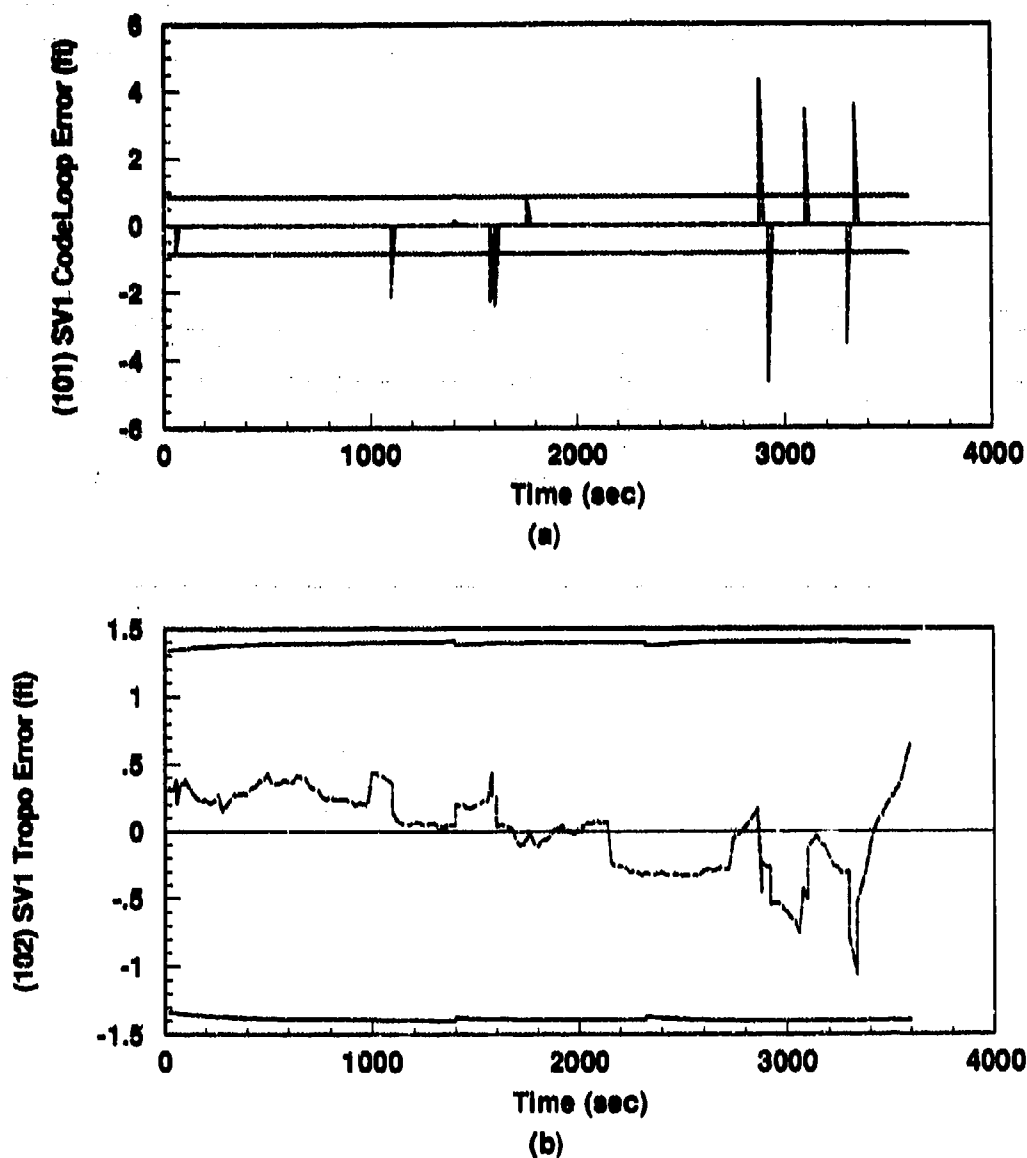


Figure I.22. Alignment: 128-State Model (a) SV 1, CodeLoop and (b) SV 1, Atmospheric Propagation Error States (Note: RRS and Both GPS Measurements)

.....	Mean Error = $\bar{M}_x - (M_x)_{true}$
.....	Mean Error $\pm \sigma_{true}$
——	$0 \pm \sigma_{filter}$

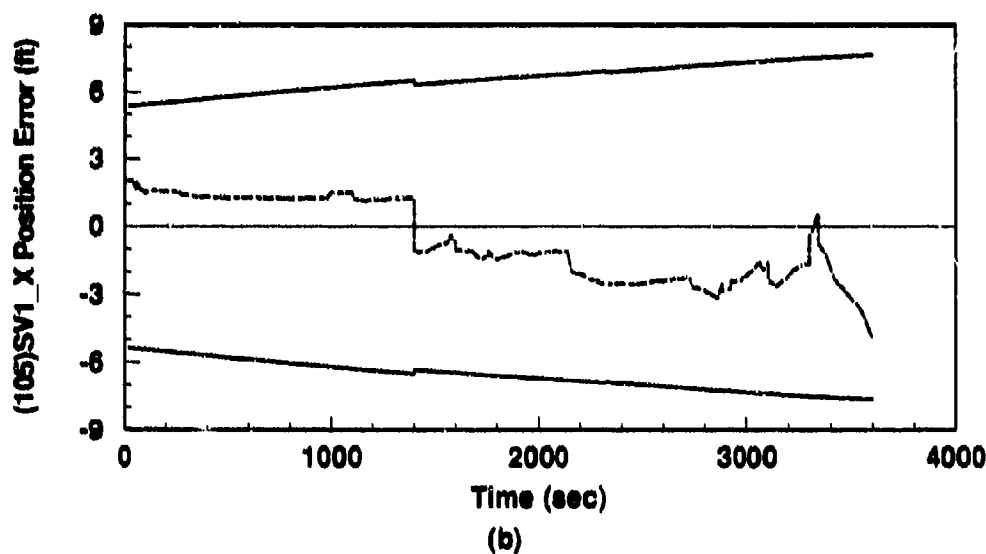
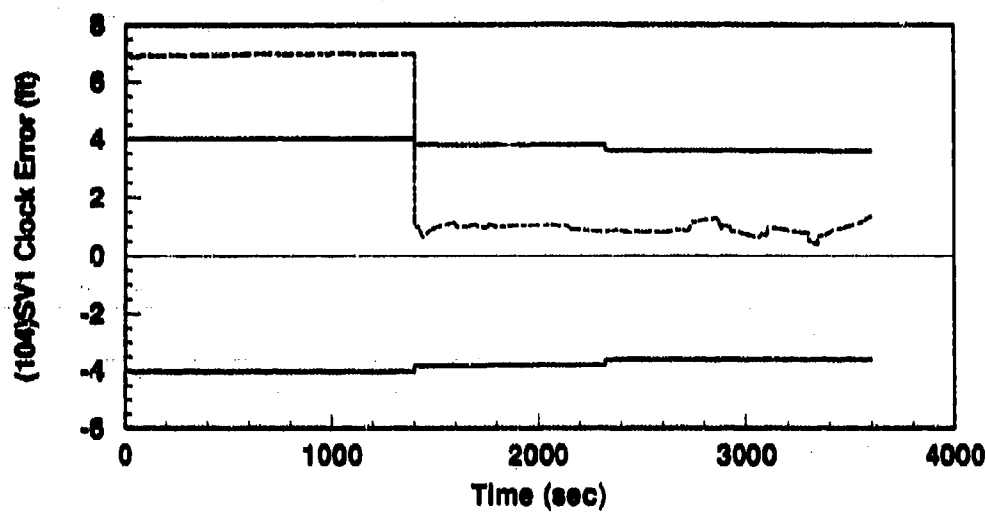


Figure I.23. Alignment: 128-State Model (a) SV 1, Clock and (b) SV 1, X Position Error States (Note: RRS and Both GPS Measurements)

----	$Mean\ Error = \bar{M}_x - (M_x)_{true}$
.....	$Mean\ Error \pm \sigma_{true}$
—	$0 \pm \sigma_{filter}$

Bibliography

1. AFWAL. *PROFGEN - A Computer Program for Generating Flight Profiles*. AFWAL-TR-76-247, Wright-Patterson AFB, OH: Avionics Laboratory, AFWAL/AAAN-2, Nov 1976.
2. Ausman, Stanley J. "Baro-Inertial Loop for the USAF Standard RLS INU," *The Institute of Navigation, National Technical Meeting* (Jan 1991).
3. Britting, Kenneth R. *Inertial Navigation Systems Analysis*. New York: Wiley-Interscience, 1971.
4. Canadian GPS Associates. *Guide To GPS Positioning*. Technical Report. Woodland Hills, CA, May 1987.
5. Carlson, Neal and Stanton Musick. *User's Manual for a Multimode Simulation for Optimal Filter Evaluation (MSOFE)*. AFWAL-TR-88-1138, Wright-Patterson AFB, OH: Avionics Lab, AFWAL/AARN-2, Apr 1990.
6. Cox Jr., D. B. "Integration of GPS with Inertial Navigation Systems," *Global Positioning System, Papers Published in Navigation: Journal of the Institute of Navigation*, pages 144-153 (1980).
7. Cunningham, Joseph R. *Performance Evaluation of GPS-Aided INS During High-Dynamic Maneuvers*. MS thesis, Air Force Institute of Technology, Wright-Patterson Air Force Base, OH, 1987.
8. Cunningham, Joseph R. and Zdzislaw H. Lewantowicz. "Dynamic Interaction of Separate INS/GPS Kalman Filters (Filter-Driving-Filter Dynamics)," *The Institute of Navigation, Proceedings of the 44th Annual Meeting* (1989).
9. DMA. *Department of Defense World Geodetic System 1984*. Defense Mapping Agency Technical Report, No. DMA TR 8350.2., Wright-Patterson AFB, OH: Defense Mapping Agency, Sep 1987.
10. Farrell, James L. *Integrated Aircraft Navigation*. New York: Academic Press, 1976.
11. GE-90D, EENG735 Class. *GPS Aided LN-93 INS*. EENG735 Class Project Final Report, Wright-Patterson AFB OH: Air Force Institute of Technology (AU), May 1990.
12. Greer, G.B. and others. "The GPS 21 Primary Satellite Constellation," *Navigation: Journal of the Institute of Navigation*, 25:9-24 (Fall 1989).
13. Hirling, James L. "Development of a Kalman Filter to Optimally Integrate a Global Positioning System Receiver and an LN-94 Inertial Navigation System." Thesis, May 1990.
14. Integrated Systems, Inc. *MATRIX, Core*. Edition 7 MDG014-010. Santa Clara, CA, Jan 1990.
15. Kumar, Muneendra. "World Geodetic System 1984: A Modern and Accurate Global Reference Frame," *Marine Geodesy*, 12:117-126 (1988).

16. Lewantowics, Zdzislaw H. Unpublished Course Notes, EENG 534, Wright-Patterson AFB OH: Air Force Institute of Technology (AU), Sep-Dec 1989.
17. Lewantowics, Zdzislaw H. Personal Interviews, Wright-Patterson AFB OH: Air Force Institute of Technology (AU), Jan-May 1990.
18. Lewantowics, Zdzislaw H. Unpublished Course Notes, EENG 635, Wright-Patterson AFB OH: Air Force Institute of Technology (AU), Jan-Mar 1990.
19. Lewantowics, Zdzislaw H. and Danny W. Keen. "Graceful Degradation of GPS/INS Performance With Fewer Than Four Satellites," *The Institute of Navigation, National Technical Meeting* (Jan 1991).
20. Litton Guidance and Control Systems. *Performance Accuracy (Truth Model/Error Budget) Analysis for the LN-93 Inertial Navigation Unit*. DID No. DI-S-21433 B/T: CDRL No. 1002. Woodland Hills, CA, Jan 1985.
21. Martin, E.H. "GPS User Equipment Error Models." In Janicsek, P.M., editor, *Global Positioning System*, Washington, D.C.: The Institute of Navigation, 1980.
22. Maybeck, Peter S. *Stochastic Models, Estimation, and Control*, Volume 1. San Diego, CA: Academic Press, 1979.
23. Maybeck, Peter S. *Stochastic Models, Estimation, and Control*, Volume 2. New York, NY: Academic Press, 1982.
24. Maybeck, Peter S. *Stochastic Models, Estimation, and Control*, Volume 3. New York, NY: Academic Press, 1982.
25. Maybeck, Peter S. Personal Interviews, Wright-Patterson AFB OH: Air Force Institute of Technology (AU), Oct-Dec 1990.
26. Maybeck, Peter S. Personal Interviews, Wright-Patterson AFB OH: Air Force Institute of Technology (AU), Jan-Mar 1991.
27. Milliken, R. J. and C. J. Zoller. "Principles of Operation of NAVSTAR and System Characteristics," *Navigation, Journal of the Institute of Navigation*, 25:3-14 (Summer 1978).
28. Pearson, Richard A. "Advance Reference System (ARS)," *Proceedings of the Twelfth Biennial Guidance Test Symposium* (Oct 1985).
29. Savage, Paul G. "Strapdown Sensors," *AGARD Lecture Series NO. 95* (June 1978).
30. Snodgrass, Faron Britt. Personal Interviews, Wright-Patterson AFB OH: Air Force Institute of Technology (AU), Jan-Mar 1990.
31. Snodgrass, Faron Britt. *Continued Development and Analysis of a New Extended Kalman Filter for the Completely Integrated Reference Instrumentation System (CIRIS)*. MS thesis, School of Engineering, Air Force Institute of Technology (AU), March 1990. AFIT/GE/90M-5.
32. Solomon, Joseph K. *CIRIS Special Study*. EENG699 Final Report, Wright-Patterson AFB OH: Air Force Institute of Technology (AU), Mar 1989.

33. Solomon, Joseph K. *Development of the Extended Kalman Filter for the Advanced Completely Integrated Reference Instrumentation System (CIRIS)*. MS thesis, School of Engineering, Air Force Institute of Technology (AU), March 1989. AFIT/GE/89M-8.
34. Widnall, William S. and Peter A. Grundy. *Inertial Navigation System Error Models*. Technical Report AFSWC TR 73-26, Cambridge, MA: Intermetrics Inc, 11 May 1973.

Vita

Captain Richard D. Stacey was born 26 June 1955 in Richlands, Virginia. After completing high school in Bristol, Tennessee he enlisted in the U.S. Air Force in September 1973. From 1974 to 1983, his assignments as an avionics technician included tours in South Carolina, Okinawa, New Jersey, and Washington state. He attended the University of Texas at Arlington under the Airman Education and Commissioning Program and received a BSEE in May of 1986. Following graduation, he completed USAF Officer Training School (OTS) and received his commission in the U.S. Air Force in August 1986. After OTS, he was assigned to the Air Force Weapons Laboratory, Kirtland AFB, NM. From September 1986 to May 1989 he was a high-energy laser systems engineer and project officer for a key Strategic Defense Initiative program known as the Large Optics Demonstration Experiment (LODE). From June 1989 to March 1991 he attended the Air Force Institute of Technology where he received the MSEE with emphasis in navigation and control systems. Following AFIT, he was assigned to the Central Inertial Guidance Test Facility, Holloman AFB, NM, where he is currently a navigation systems analyst.

Permanent address: 208 East Road
Bristol, Tennessee 37620